Rockwell Automation Application Content

Device Libraries

Reference Manual

Kinetix 5100 Device / Operation Objects

raC_Dvc_K5100 v2.0
raC_Opr_K5100_MSO v2.0
raC_Opr_K5100_MSF v2.0
raC_Opr_K5100_MAJ v2.0
raC_Opr_K5100_MAT v2.0
raC_Opr_K5100_MAM v2.0
raC_Opr_K5100_MAH v2.0
raC_Opr_K5100_MAG v2.0
raC_Opr_K5100_MAS v2.0
raC_Opr_K5100_MAFR v2.0
raC_Opr_K5100_MAI v2.0

Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://literature.rockwellautomation.com) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances or death, property damage, or economic loss. Attentions avoid a hazard, and recognize the consequence.
SHOCK HAZARD	Labels may be on or inside the equipment, that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Allen-Bradley, Rockwell Automation, and TechConnect are trademarks of Rockwell Automation, Inc.Trademarks not belonging to Rockwell Automation are property of their respective companies.

Table of Contents

1. Overview 5

1.1 Prerequisites 6
1.1.1 Required Software 6
1.1.2 Optional Software 6
1.1.3 Required Hardware 6
1.1.4 Optional Hardware 6
1.1.5 Required Configuration using KNX5100C 6

1.2 Functional Description 7
1.2.1 The Kinetix 5100 Preconfigured Device Objects: 7
1.2.2 The Kinetix 5100 Operational Objects: 7

1.3 Footprint 7
1.3.1 raC_Dvc_K5100 7

2. Device Object Architecture 8

2.1 Functional Requirements 8
2.1.1 Physical functions 8
2.1.2 Virtual functions 8
2.1.3 Secondary functions 9

3. Data Types 10

3.1 raC_UDT_Itf_K5100_Cfg 10

3.2 Example configuration with Position Units 11

3.3 raC_UDT_Itf_K5100_Set 11

3.4 raC_UDT_Itf_K5100_Cmd 13

3.5 raC_UDT_Itf_K5100_Sts 15

3.6 raC_UDT_Event 16

3.7 raC_UDT_LookupMember_STR0082 17

4. Motion Operation AOI's 18

4.1 raC_Opr_K5100_MSO - Motion Axis Servo On 19

4.2 raC_Opr_K5100_MSF - Motion Axis Servo OFF 20

4.3 raC_Opr_K5100_MAJ - Motion Axis Jog 21

4.4 raC_Opr_K5100_MAT - Motion Axis Torque 23

4.5 raC_Opr_K5100_MAM - Motion Axis Move 25
4.5.1 Set_MoveType 26
4.5.2 Set_PositionOverride 27
4.5.3 Set_PositionCommandOverlap 27

4.6 raC_Opr_K5100_MAH - Motion Axis Home 29

4.7 raC_Opr_K5100_MAG - Motion Axis Gear and E-Gear Ratio 31

4.8 raC_Opr_K5100_MAS - Motion Axis Stop 33

4.9 raC_Opr_K5100_MAFR - Motion Axis Fault Reset 36

4.10 raC_Opr_K5100_MAI - Motion Axis Index 37

4.11 Common Error Codes 38

5. State Model 40

5.1 State Definitions 41

5.2 State Actions 41

6. Application Code Manager 42

6.1 Control Asset Objects 42

6.2 Implement Objects 43

6.2.1 raC_LD_Dvc_K5100 43

7. User Interface 46

7.1 Faceplate Navigation: 46

7.2 Common Status 48

7.3 Home Screen 49

7.4 I/O Screen 50

7.5 Trend Screen 51

7.6 Configure Screen 52

7.7 Fault Warning Screen 53

7.7.1 Details Screen 53

8. Appendix 54

1. Overview

When the Kinetix 5100 drive is configured for IO mode operation and used with Studio 5000 Logix Designer, the use of the pre-defined AOP (Add On Profile) and AOI (Add On Instruction) objects provide an organized and robust way to program your simple motion control application.

The Kinetix 5100 has been designed to use the AOI objects since its launch with major firmware revision 1.xx. These objects have evolved to include the Device Object AOI (described below) which provides a robust method to handle the interface with the device I/O assemblies.

This Device Object AOI provides many advantages including:

- Eliminating the need for synchronous copy and other correctly placed interlocks, which were sometimes not obvious, to make the Motion Operation AOI's operate as intended.
- Robust operation that includes the interface of drive command/status within the Motion Operation AOI's
- Designed for integration with the Power Device Library and Machine Builder Libraries and its framework for programming.
- An HMI faceplate (optional) that is used with the Device Object AOI (required) to provide the status information as well as simple control of the drive using View Designer or FactoryTalk View for programming.
- The creation of Position Units. This allows the Kinetix 5100 to use native units of drive counts or to use Position units using a Conversion Constant and Motion Resolution.

The table below shows the differences between the AOI libraries and the use of the Device Object AOI.

Kinetix Firmware Major Rev	Device Object AOI used	AOI Library Naming
1.xx	NO	raC_Dvc_K5100_xxx
2.xx	YES	raC_Opr_K5100_xxx

If using the Kinetix 5100 with major revision 2.xx, the Device Object and Motion Operation AOI's must be used. AOI Versions 1.xx and 2.xx are not interchangeable and cannot be combined to perform motion control. The AOI library containing raC_Dvc instructions will continue to be available for legacy applications. They can be downloaded from the PCDC. Any new applications that will be using the Kinetix 5100 with major revision 2.002 or later should use the Device Object AOI and the new motion operation AOI's described in this document. Each Kinetix 5100 drive will require a unique instance of the Device Object AOI.

The Kinetix 5100 Power Motion Device Objects include Add-On Instruction definitions, rung instances, and an HMI faceplate providing:

- Detailed Device Data Collection and Delivery
- Enhanced Device Status and Diagnostics
- Common Control Interfaces maximizing flexible Automation Device selection &

application code reuse Device Object Use Cases:

- Basic Device Maintenance and Diagnostics
- Virtual Device Operations for startup and commissioning

1.1 Prerequisites

1.1.1 Required Software

- Studio5000 Logix Designer v31.00.00 and later for PAC application development
- KNX5100C software

1.1.2 Optional Software

- Studio5000 View Designer v4.01 and later for PanelView 5000 application development (when using the faceplate screens)
- FactoryTalk View Studio v10 and later for PanelView Plus Application Development (when using the faceplate screens)

1.1.3 Required Hardware

At minimum, the following hardware is required to utilize these instructions and faceplates:

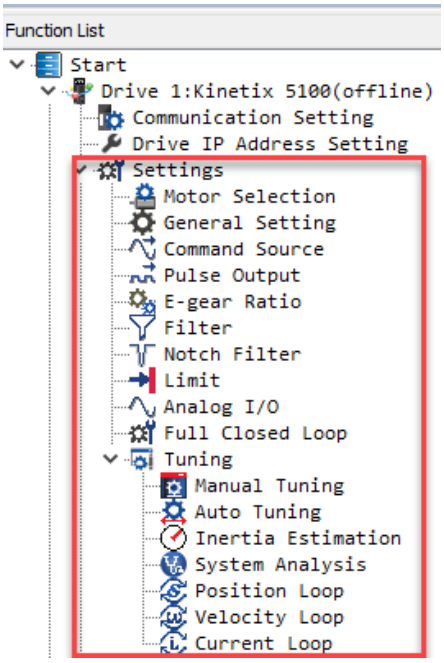
- Logix controller with V31.001 or later firmware
- Kinetix 5100 drive with V2.001 or later firmware

1.1.4 Optional Hardware

- PanelView 5500 operator interface with V4 or later firmware (when using the faceplate screen
- PanelView Plus operator interface (when using the faceplate screen)

1.1.5 Required Configuration using KNX5100C

The Kinetix 5100 requires initial configuration of the drive using KNX5100C software. Besides changing the drive's IP address and operation mode, this initial configuration includes a review of the Function List>Settings. The settings within these categories will need to be modified depending on your hardware and your application. The categories that require attention are shown here:



A minimal configuration is shown in the 'How To' video that came with this package. KNX5100C software may be downloaded from the Product Compatibility and Download Center. Search for "Kinetix 5100".

https://compatibility.rockwellautomation.com/Pages/home.aspx

1.2 Functional Description

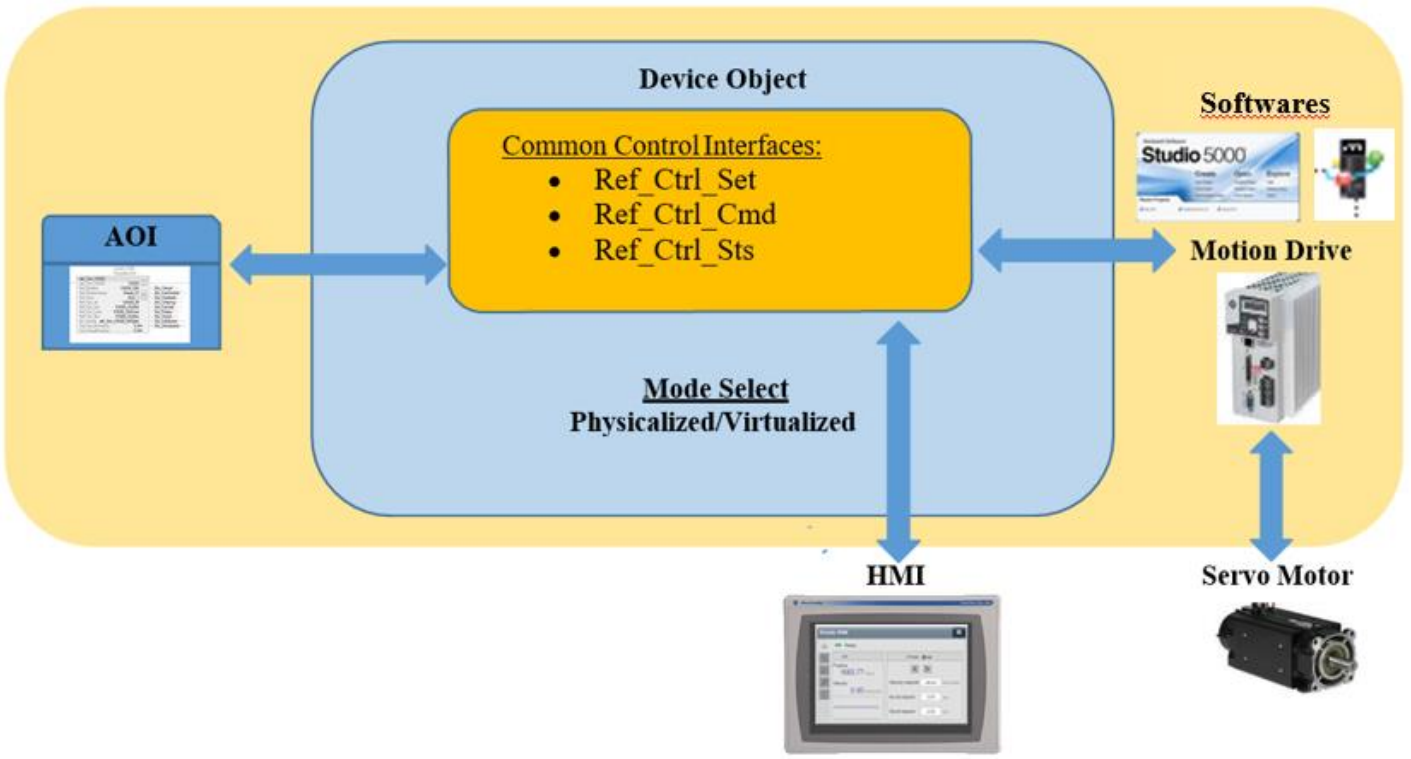
1.2.1 The Kinetix 5100 Preconfigured Device Objects:

Collect, process, and deliver data between devices and application logic.
Provide device status & diagnostics faceplates for machine startup, commissioning, operations, and maintenance.
Include common HMI faceplate components, device states, and control interfaces providing application development and operation efficiency.

1.2.2 The Kinetix 5100 Operational Objects:

The Operational Objects are listed in Section 4

These features deliver the user a tool that provides quick feedback, shortens recovery time, and simplifies implementation.



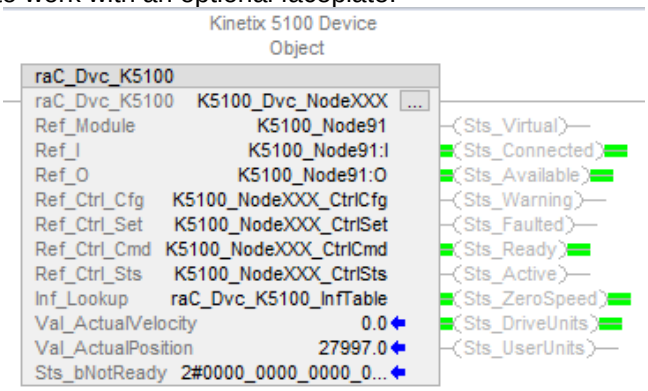
1.3 Footprint

1.3.1 raC_Dvc_K5100

Characteristic	Description	Value	Unit
Definition	Estimated memory required to store the object definition, including all dependents (including 1 Instance).	166	kB
Instance	Estimated memory required per object instantiated.	3	kB
Execution L7x	Estimated execution time / scan footprint evaluated in 1756-L7x PAC	230	µs

2. Device Object Architecture

The Device Object handler is used to interface between the drive and the module assemblies. It is also designed to work with an optional faceplate.



Example of the Device Object Handler for the Kinetix 5100 drive

2.1 Functional Requirements

2.1.1 Physical functions

The following functions are applied when device object is selected as physical.

Activate: This command will activate the drive power structure. This means the motor is energized. The Kinetix 5100 can perform motion when the drive state is Active.

Deactivate: This command will deactivate the drive power structure. This means the motor is disabled. The motor cannot perform motion when the drive is state is not Active.

Reset Fault: Command to reset faults on the device and reset status bits in device object. Command can be initiated from control interface and HMI interface.

Start Motion: A zero-to-one transition means the motion command is issued from the external controller. There are additional settings that must be made in conjunction with this bit. These settings are embedded in the AOI Objects.

Stop Motion: A zero-to-one transition will stop any currently executing motion and leave the drive activated.

2.1.2 Virtual functions

The following functions are applied when device object is selected as virtual.

Activate: Set status to reflect successful activate command. Set status output to reflect frequency setpoint, drive-related signals like Logic Command and Out Reference are not altered.

Deactivate: Reset the active status and actual speed status.

Jog: This function is accessible from the HMI faceplate only. When Jog command is set, activate drive power structure, and speed reference will follow Jog setpoint. Once the command is reset, deactivate the drive power structure by issuing Stop command. Individual jog forward and jog reverse commands are used to jog in either direction. Unlike other command bits, this command is not unlatched by the device object.

Fault Reset: Command to reset status bits in device object. Command can be initiated from control interface and HMI interface.

2.1.3 Secondary functions

- **First Warning:** This function capturing the first warning triggered in the Device Object and displays the respective description in faceplate. This warning contains the timestamp and can be logged and stored by the user.
- **First Fault:** Capture the first fault from the device. Display the respective description in faceplate. This fault is captured in the Device Object with the timestamp and can be logged by the user.
- **Event log:** The queue contains fault code, description, and time stamp. When the faceplate is used, a log of 4 events is captured. These can be displayed using the faceplate.
- **Status Display:** Display device status. Additional details are provided in the next section on the individual status values.

3. Data Types

The following Power Motion Common Control Interface tags are the primary device program tags to read from and write to when interfacing to power motion devices.

Note: For further information and examples on how to interface the power device objects with your specific application code refer to the “How_To_Interface_with_Power_Device_Logix.mp4” video within the General Documents folder of the Power Device Library Download files.

The Power Device Library may be downloaded from the Product Compatibility and Download Center. Search for “Power Device Library”.
https://compatibility.rockwellautomation.com/Pages/home.aspx

3.1 raC_UDT_Itf_K5100_Cfg

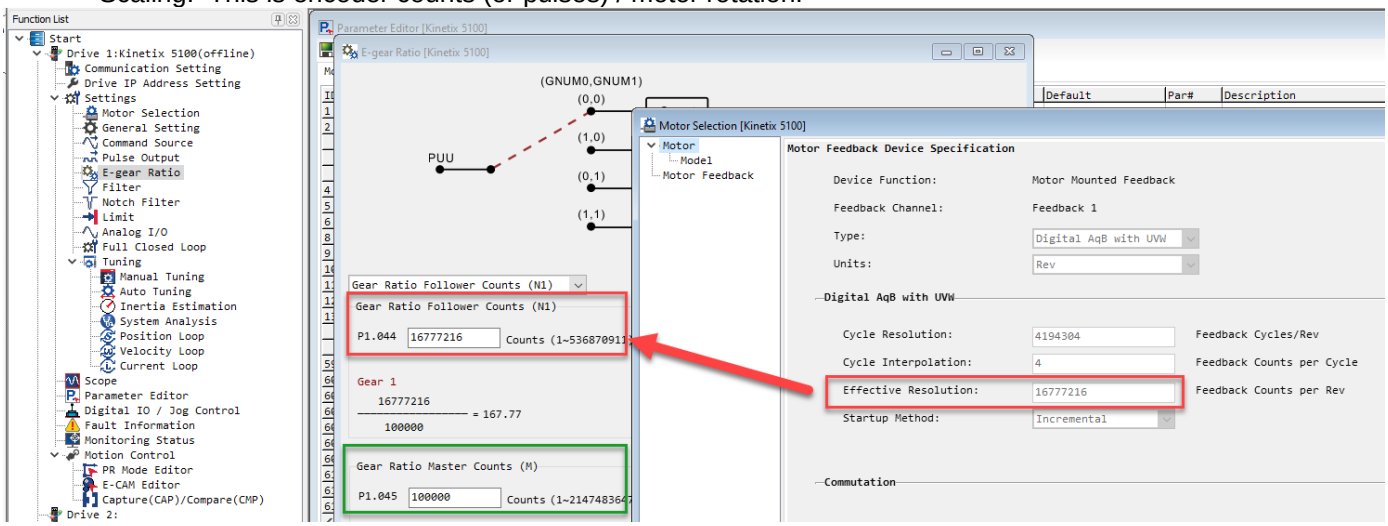
raC_UDT_Itf_K5100_Cfg is the Power Motion Common Control Interface User-Defined Data Type for device configuration. Its members provide selection between Drive units (counts) or User units with settings for user units.

This is very useful because the Kinetix 5100 natively supports only drive units. When the Operating Units = 1, the Motion Resolution and ConversionConstant values are used. Position Scaling originates from the KNX5100C software and is used together with the Cfg tags here to derive user scaling units.

Member	Description	DataType
OperatingUnits	0 = Drive Units; 1 = UserUnits	DINT
MotionResolution	Motion Counts per Motor Revolution	DINT
ConversionConstant	Motion Counts per Position Unit	REAL

3.2 Example configuration with Position Units

The E-Gear ratio (KNX5100C>Function List>E-Gear Ratio) is always used to provide a representation of positioning (units or counts) or to define a Pulse-Pulse Following relationship (MAG/PT). When the E-Gear ratio is changed, the positioning of the axis is changed. When not using the MAG AOI, the E-Gear ratio is used to define position scaling.
When Operating Units =1, Position Units are used, and we can define application units instead of using drive counts. In KNX5100C software, the E-Gear ratio is defined to provide Position Scaling. This is encoder counts (or pulses) / motor rotation.



All Position Unit configurations must:

- Configure *P1.044 Gear Ratio Follower Counts* to be the same as the motor feedback resolution.
- Configure *P1.045 Gear Ratio Master Counts* to provide motor feedback counts / motor rotation.
- This value is user defined and can be any count value, default values with high-resolution encoders are 100,000 counts/motor rotation. The E-Gear configuration is used with the Device Object Cfg tags.

K5100_NodeXXX_CtrlCfg	{...}	{...}	Automation Device Interface
K5100_NodeXXX_CtrlCfg.OperatingUnits	1		Automation Device Interface 0 = Drive Units; 1 = UserUnits
K5100_NodeXXX_CtrlCfg.MotionResolution	100000		Automation Device Interface Motion Counts per Motor Revolution
K5100_NodeXXX_CtrlCfg.ConversionConstant	100000.0		Automation Device Interface Motion Counts per Position Unit

The Device Object Cfg values must:

- Set *Cfg.MotionResolution* = *P1.045 Gear Ratio Master Counts* –> Motion Counts / Motor Revolution
- Set *Cfg.ConversionConstant* based on the Counts/Position Unit -> Motion Counts / Position Unit that is required for your application.

The example above will result in Position Units = motor rotations. Now, entry values that originally used drive counts can be entered as motor rotations.

3.3 raC_UDT_Itf_K5100_Set

raC_UDT_Itf_K5100_Set is the Power Motion Common Control Interface User-Defined Data Type for device settings. Its members provide application program access to allow or inhibit commands and settings from the device faceplate or other external sources. The table below shows member names, descriptions, and tag data types.

For example, to inhibit write commands from the device faceplate or other external sources write a 1 to the *ModuleName_AOI*_CtlrSet.InhibitCmd program tag from your

application program. This would prevent a jog command from the device faceplate.

Member	Description	DataType
bInhibit	Bit overlay for external access restriction	DINT
InhibitCmd	1 = Inhibit user Commands from external sources; 0 = Allow Control. This is only used with the optional device faceplate.	BOOL
InhibitSet	1 = Inhibit user Settings from external sources; 0 = Allow This is only used with the optional device faceplate.	BOOL
OperatingMode	Determines the drive operating mode when “Start Motion” has a zero-to-one transition. 1 - Position mode 2 - Speed mode 3 - Home mode 4 - Torque mode 5 - Gear mode (Fixed Ratio, based on present E-Gear ratio) 6 - Index mode 7 – Reserved 8 – Gear Mode (Variable Ratio, based on Master/Slave tag values) 9 – Enhanced MAT mode	DINT
MoveType	Specify the type of move. 0 = Absolute 1 = Incremental 2 = Rotary Shortest Path 3 = Rotary Positive 4 = Rotary Negative 7 = Relative 8 = Capture	DINT
PositionCommandOverlap	Allows overlapping of successive movements.	BOOL
PositionCommandOverride	Allows interruption of current movement, replacing it with a new movement.	BOOL
CapturedPositionSelect	Capture position selection (First capture or second capture).	BOOL
Position	Determines the command position.	REAL
Velocity	Determines the command speed.	REAL
Accel	Determines the command acceleration.	REAL
Decel	Determines the command deceleration.	REAL
Torque	Determines the command torque.	DINT
TorqueRampTime	Determines the command torque ramp time.	DINT
StartingIndex	This entry is the PR (Position Register) the drive should execute.	DINT
HomingMethod	Homing Method.	DINT
HomeReturnSpeed	Determines the command home return speed.	REAL
CamMasterReference	Future: Determines the master position reference of	DINT

	CAM.	
CamExecutionSchedule	Future: Determines the method used to execute the CAM profile.	DINT
CamExecutionMode	Future: Determines if the cam profile is executed only one time or repeatedly.	DINT
CamStopMode	Future: Determines the stop mode of CAM.	BOOL
CamSlaveScaling	Future: Scales the total distance covered by the slave axis through the cam profile.	DINT
CamLockPosition	Future: Determines the starting location in the cam profile	DINT
CamMasterLockPosition	Future: Determines the master location where the slave axis locks to the mater axis.	DINT
CamMasterLeadingCounts	Future: Determines the leading counts (master axis) before the cam profile is executed.	DINT
CamMasterUnlockCounts	Future: Determines the unlock counts (master axis) when the cam profile is executed.	DINT
CamMasterCyclicLeadingCounts	Future: Determines the cyclic leading counts (master axis) during the cam profile is executed.	DINT
GearRatioSlaveCounts	Integer value representing slave counts. This value is P1.044 Gear Ratio Follower Counts from the E-Gear ratio in KNX5100C software.	DINT
GearRatioMasterCounts	Integer value representing master counts. This value is P1.045 Gear Ratio Master Counts from the E-Gear ratio in KNX5100C software.	DINT

3.4 raC_UDT_Itf_K5100_Cmd

raC_UDT_Itf_K5100_Cmd is the Power Motion Common Control Interface User-Defined Data Type for device commands. Its members provide application program access to common basic device commands.

The table below shows member names, descriptions, and tag data types.
All the commands are available whether operating the device physically or virtually.

While it is possible, it is not typical to modify any of these UDT values directly. The Motion Operation AOI's will manipulate these values as a result of their operation.

Member	Description	DataType
bCmd	Commands (Bit Overlay)	DINT
Physical	1 = Operate as a physical device	BOOL
Virtual	1 = Operate as a virtual device	BOOL
ResetWarn	1 = Reset device warning	BOOL
ResetFault	1 = Reset device trip or fault	BOOL
Activate	1=Activate Output Power Structure	BOOL
Deactivate	1=DeActivate Output Power Structure	BOOL
StartMotion	A zero-to-one transition means the motion command is issued from the external controller.	BOOL

StopMotion	A zero-to-one transition will stop any active motion command in the drive.	BOOL

3.5 raC_UDT_Itf_K5100_Sts

raC_UDT_Itf_K5100_Sts is the Power Motion Common Control Interface User-Defined Data Type for device status. Its members provide application program access to device states, status, and diagnostic data. The table below shows member names, descriptions, and tag data types.

Input	Description	DataType
eState	Enumerated state value: 0=Unused 1=Initializing 2=Disconnected 3=Disconnecting 4= Connecting 5=Idle 6=Configuring 7=Available	DINT
FirstWarning	Capture the First Alarm Bit to trigger. Display the respective Description and Time Stamp on Faceplate. Log the same in Event Queue.	raC_UDT_Event
FirstFault	Capture the Fault Code of the device. Display the respective code, description, and timestamp on faceplate. Log the same in Event Queue.	raC_UDT_Event
eCmdFail	Enumerated command failure code	DINT
bSts	Status (Bit Overlays)	DINT
Physical	1 = Operating as a physical device	BOOL
Virtual	1 = Operating as a virtual device	BOOL
Connected	1 = PAC to device connection has been established.	BOOL
Available	1 = The automation device is available for interaction with the user program	BOOL
Warning	1 = A warning is active on the automation device	BOOL
Faulted	1 = A fault is active on the automation device	BOOL
Ready	1=Device is ready to be Activated	BOOL
Active	1= Device power structure is active	BOOL
ZeroSpeed	1=Motor is within zero speed tolerance (this tolerance is defined in KNX5100C software)	BOOL
Homed	Indicate whether the drive completed the home operation.	BOOL
AtReference	Depending on the motion command (position, speed, torque), AtReference will be 1 when the actual reference = command reference	BOOL
CommandInProgress	Toggles state when a motion command is active in the drive. This bit changes state (toggles between 0 & 1) when a new command is executed from the drive. **IMPORTANT:** once this bit changes state, it remains in that state for the duration of the command; it will toggle to the opposite state (and remain in that state) once a new command is received.	BOOL
FaultCode	Active Fault Code in the drive	DINT

WarningCode	Active Warning Code in the drive	DINT
OperatingMode	Indicate which operating mode is currently used.	DINT
MotorType	Indicate which type of motor is connected to the drive. Rotary Motor = 1, Linear Motor =2 (Future)	DINT
ActualPosition	Actual position of the motor. Units depend on the Cfg settings. These can be drive counts or Position Units.	REAL
ActualVelocity	Actual speed of the motor. Units depend on the Cfg settings. These can be 0.1 RPM/sec or Position Units.	REAL
ActualTorque	When the operating mode is 4, Torque Mode, this represents the % motor torque.	REAL
ActiveIndex	Indicates the currently executing Position Register PR)	DINT
ParameterMonitor1Value	Parameter monitor variable 1 value. These are Parameter ID's mapped using Function List>Parameter Editor>Status Monitor in KNX5100C software	DINT
ParameterMonitor2Value	Parameter monitor variable 2 value. These are Parameter ID's mapped using Function List>Parameter Editor>Status Monitor in KNX5100C software	DINT
ParameterMonitor3Value	Parameter monitor variable 3 value. These are Parameter ID's mapped using Function List>Parameter Editor>Status Monitor in KNX5100C software	DINT
ParameterMonitor4Value	Parameter monitor variable 4 value. These are Parameter ID's mapped using Function List>Parameter Editor>Status Monitor in KNX5100C software	DINT
ParameterMonitor5Value	Parameter monitor variable 5 value. These are Parameter ID's mapped using Function List>Parameter Editor>Status Monitor in KNX5100C software	DINT

3.6 raC_UDT_Event

raC_UDT_Event is an array of size 4 is to be used to log the FirstWarning and FirstFault capture. The data is captured in FIFO order. The same should be displayed on the faceplate. This UDT is created as part of the Device Object, however, is only used with the optional faceplate. This can be assigned by the user to record events in Logix. When using Machine Builder Libraries, a suite of instructions is available to customize event handling.

Member	Description	DataType
Type	1 = Status 2 = Warning 3 = Fault 4 ...n = User	DINT
ID	User definable event ID	DINT
Category	User definable category (Electrical, Mechanical, Materials, Utility, etc.)	DINT
Action	User definable event action code	DINT
Value	User definable event value or fault code	DINT
Message	Event message text	STRING
EventTime_L	Timestamp	LINT
EventTime_D	Timestamp (Y,M,D,h,m,s,us)	DINT[7]

3.7 raC_UDT_LookupMember_STR0082

Member	Description	DataType
Code	Stores the value of device fault code.	DINT
Desc	Stores the Messages related to fault code.	STRING

4. Motion Operation AOI's

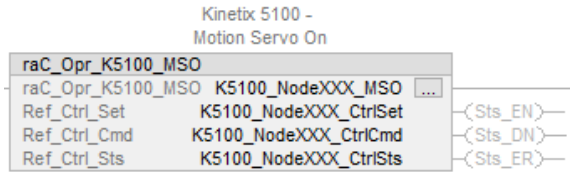
These Motion Operation AOI's are designed to simplify the programming for your motion application. Consider that the Kinetix 5100 is not an Integrated Motion on Ethernet/IP (CIP) drive and is not part of the Motion Group. The drive communication (Class 1 Ethernet/IP) for these AOI's is based on the RPI of your Kinetix 5100 in the Add-On Profile. Typical values for this communication rate are 20ms with a maximum of 100ms.

IMPORTANT: While these AOI's were developed to look and operate similarly to CIP motion instructions, **they do not behave the same as instructions from the CIP Motion Library**. The Kinetix 5100 is not an Integrated Motion on Ethernet/IP (CIP) motion drive.

Member	Description
raC_Opr_K5100_MSO	Motion Axis Servo On. Use the Motion Servo On instruction to activate the motor.
raC_Opr_K5100_MSF	Motion Axis Servo Off. Use the Motion Servo Off instruction to deactivate the drive output for the specified axis and to deactivate the axis servo loop
raC_Opr_K5100_MAJ	Motion Axis Jog Use the Motion Axis Jog instruction to move an axis at a constant speed until you tell it to stop.
raC_Opr_K5100_MAT	Motion Axis Torque Use the Motion Axis Torque instruction to move an axis at a constant torque with the speed limit.
raC_Opr_K5100_MAM	Motion Axis Move Use the Motion Axis Move instruction to move an axis to a specified position.
raC_Opr_K5100_MAH	Motion Axis Home Use the Motion Axis Home instruction to home an axis.
raC_Opr_K5100_MAG	Motion Axis Gear Use the Motion Axis Gear instruction to set the gear ratio between the PUU (Position of User Unit) and encoder counts and specify the acceleration rate during gear ratio change for the position control in the Kinetix 5100 drive.
raC_Opr_K5100_MAS	Motion Axis Stop Use the Motion Axis Stop instruction to stop a specific motion process on an axis or to stop the axis completely.
raC_Opr_K5100_MAFR	Motion Axis Fault Reset Use the Motion Axis Fault Reset instruction to clear some motion faults for an axis. Other faults cannot be cleared until you power cycle the drive. The faults, which can be cleared by raC_Opr_K5100_MAFR, are listed in Fault list section.
raC_Opr_K5100_MAI	Motion Axis Index Use the Motion Axis Index instruction to execute the index (PR) function of the drive. This can be a Position Index or one of the other PR types available in the drive.

4.1 raC_Opr_K5100_MSO - Motion Axis Servo On

Use the Motion Servo On instruction to activate the motor. This instruction must be used while there are no active faults on the drive and the drive is in a Ready State.

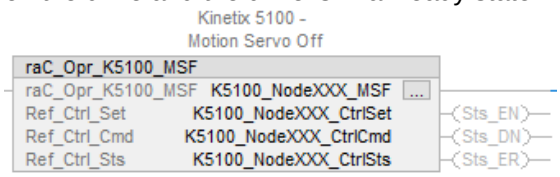


Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and remains set as the message transaction to activate the drive is initiated and in process. It remains set while the rung-in condition is true and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the cmd to activate the drive is being acknowledge.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI

This example shows the Motion Operation AOI (raC_Opr_K5100_MSO). The Device Object status bits should be used in your application logic, when possible, for Kinetix 5100 status instead of using the specific instance bits: in this case Sts_EN/DN. The Device Object Sts.Ready state checks additional states like valid Ethernet/IP communication with the drive.

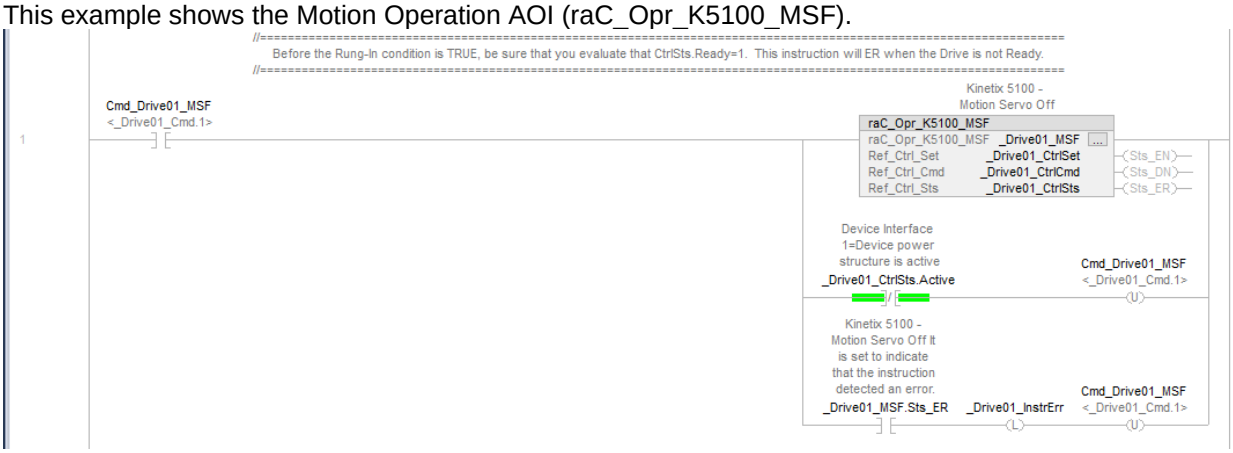
4.2 raC_Opr_K5100_MSF - Motion Axis Servo OFF

Use the Motion Servo Off instruction to de-activate the motor. This instruction must be used while there are no active faults on the drive and the drive is in a Ready state.



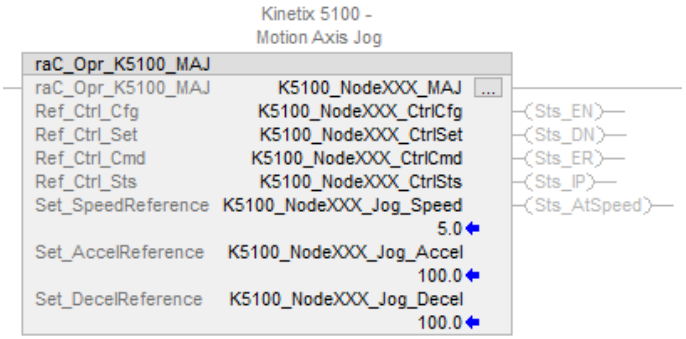
Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and remains set as the message transaction to de-activate the drive is initiated and in process. It remains set while the rung-in condition is true and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to de-activate the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI

This example shows the Motion Operation AOI (raC_Opr_K5100_MSF).



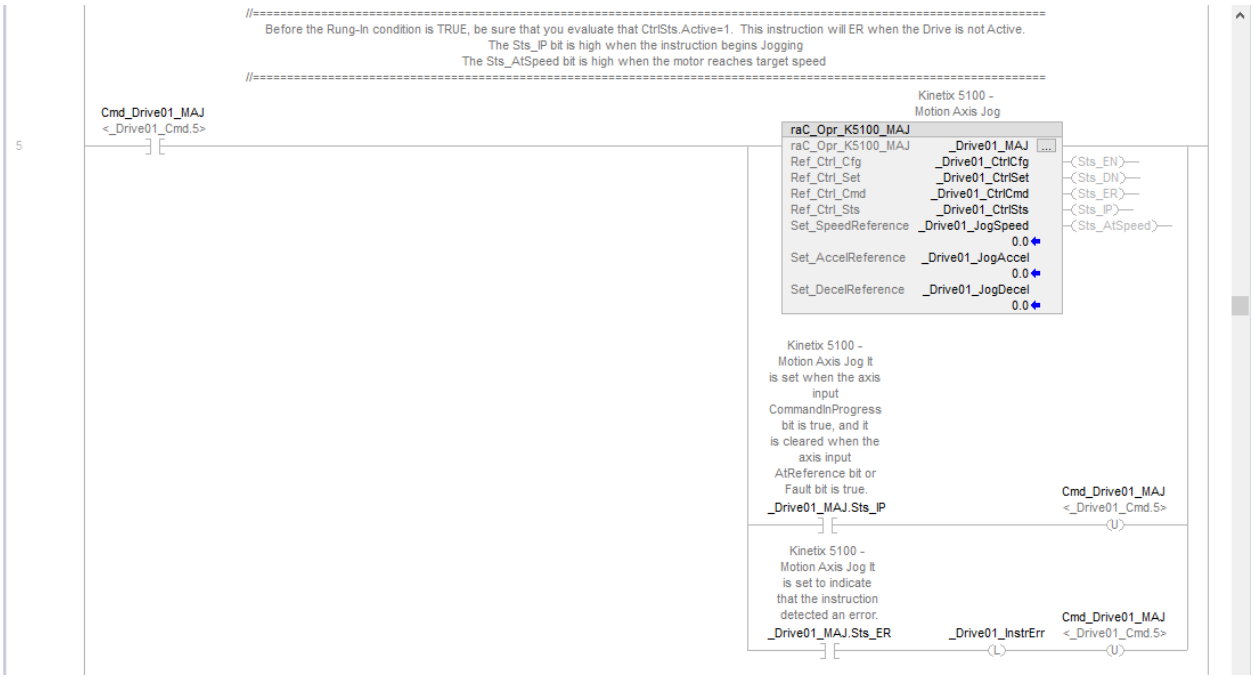
4.3 raC_Opr_K5100_MAJ - Motion Axis Jog

Use the Motion Axis Jog to accelerate or decelerate the motor to a constant speed without termination.



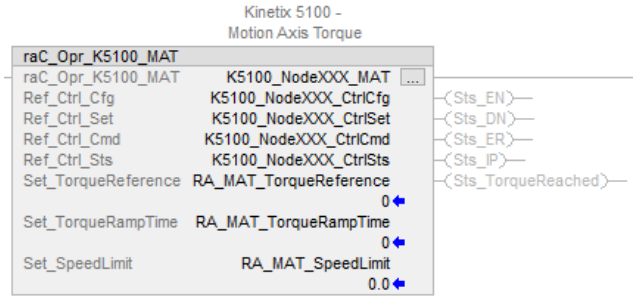
Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to Jog is initiated and in process. It remains set while the rung-in condition is true and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to Jog the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the Jog message transaction is successful, and the motor begins to move. This bit will remain set as the motor is moving towards the target speed (Accel or Decel). It remains set while the Jog is active, regardless of the rung-in condition
Sts_AtSpeed	BOOL	This bit is set when the rung makes a false-to-true transition, the Sts_IP is set, and the Target Speed is reached. This bit will remain set while the Jog is active and AtSpeed condition is true.
Ref_Ctrl_Cfg	raC_UDT_Ift_K5100_Cfg	Use the associated Device Object Configuration control interface for this AOI
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI
Set_SpeedReference	REAL	Target speed for the jog. Units: 0.1 RPM, Range: +/- 80,000
Set_AccelReference	REAL	Accel rate used for the jog. Units: 0.1 RPM/s, Range:458…30,000,000
Set_DecelReference	REAL	Decel rate used for the jog. Units: 0.1 RPM/s, Range:458…30,000,000

This example shows the Motion Operation AOI (raC_Opr_K5100_MAJ). The rung-in condition is reset once the motor is moving.



4.4 raC_Opr_K5100_MAT - Motion Axis Torque

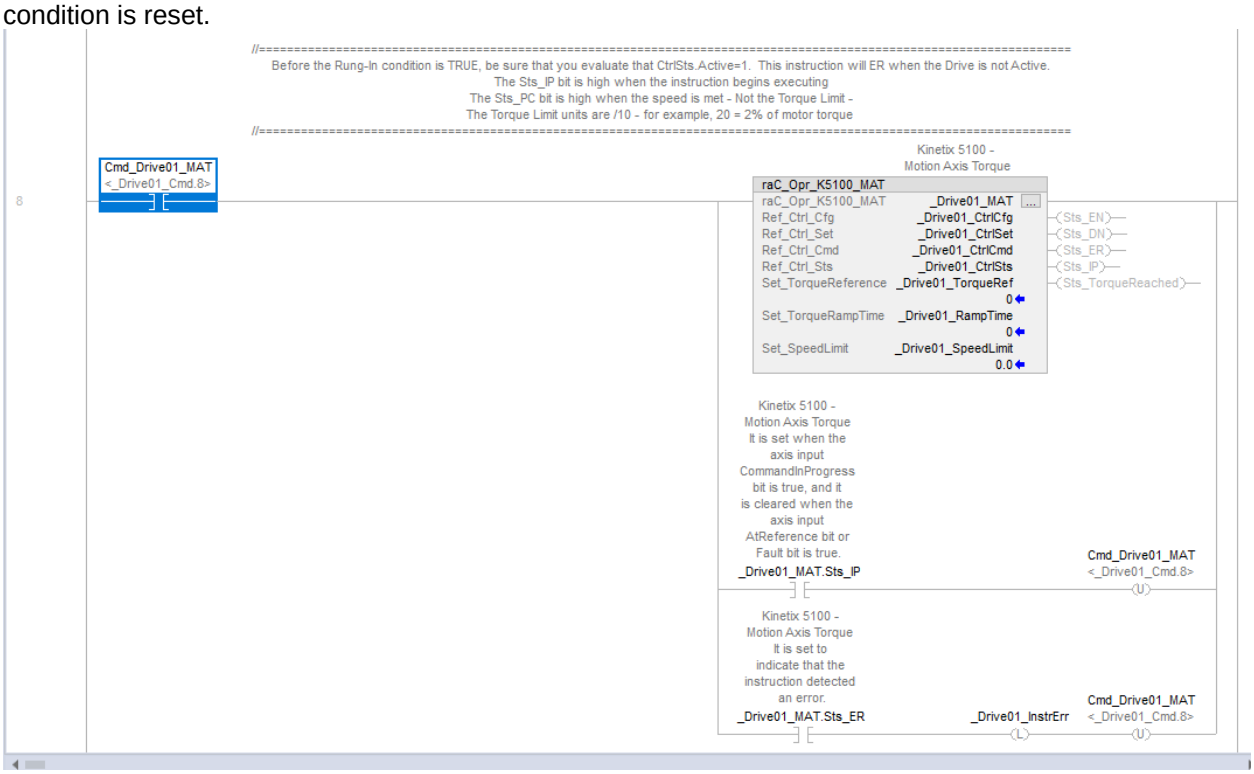
The Motion Axis Torque instruction allows you to use torque limiting while a pre-defined speed is used to move the motor. The first time the pre-defined torque limit is reached, the Sts_TorqueReached bit is set. While the Sts_TorqueReached bit is set, the MAT operation remains active until it is terminated by an raC_Opr_K5100_MAS (Motion Axis Stop)/MSF (Motion Axis Servo Off), or fault of some kind. The torque and speed entries are bi-directional.



Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and remains set as the message transaction to execute the MAT is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the MAT message transaction is successful, and the motor begins to move. This bit remains set while the MAT operation is active.
Sts_TorqueReached	BOOL	This bit is set when the rung makes a false-to-true transition, the Sts_IP is set, and the Set_TorqueReference value is reached. This bit is set (and remains set) on the first occurrence of this condition.
Ref_Ctrl_Cfg	raC_UDT_Ift_K5100_Cfg	Use the associated Device Object Configuration control interface for this AOI
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI
Set_SpeedReference	REAL	Target speed for the jog. Units: 0.1 RPM, Range: +/- 80,000
Set_TorqueReference	DINT	Torque Limit used. Units: 0.1 % motor torque, Range: +/- 4,000

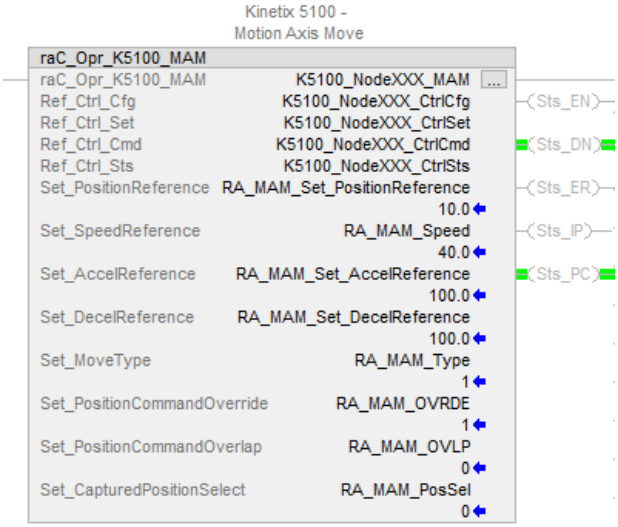
Set_TorqueRampTime	DINT	The time to reach the Torque Limit. Units: ms; Range: 1…65,500
Set_SpeedLimit	REAL	Speed Limit during the MAT operation. Units: 0.1 RPM. Range:=/-80,000

This example shows the Motion Operation AOI (raC_Opr_K5100_MAT). Once Sts_IP is set, the rung-in condition is reset.



4.5 raC_Opr_K5100_MAM - Motion Axis Move

Use the Motion Axis Move to execute an index based on the instruction's configuration.

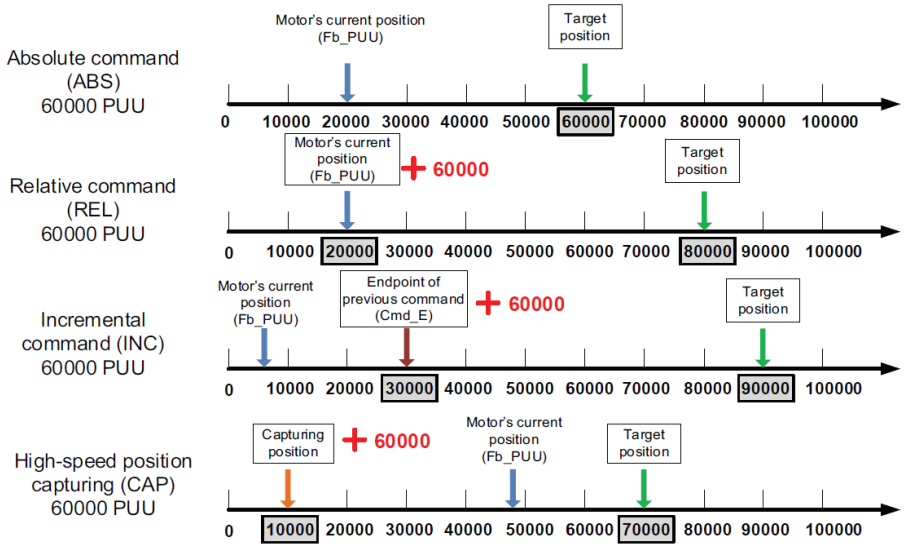


Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to Index is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to Index the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the Index message transaction is successful, and the motor begins to move. This bit will remain set as the motor is executing the index.
Sts_PC	BOOL	Process Complete. This bit is set when the rung makes a false-to-true transition, the Sts_IP is set, and the Target Position is reached.
Ref_Ctrl_Cfg	raC_UDT_Ift_K5100_Cfg	Use the associated Device Object Configuration control interface for this AOI
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI

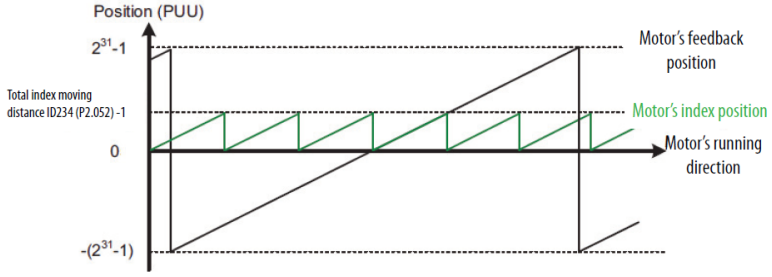
Set_SpeedReference	REAL	Target speed for the MAM. Units: 0.1 RPM, Range: +/-80,000; Position Units are converted into RPM using the Device Object AOI
Set_AccelReference	REAL	Accel rate used for the Index. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI
Set_DecelReference	REAL	Decel rate used for the Index. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI
Set_MoveType	INT	Specify the type of index. 0=Absolute 1=Incremental 2=Rotary Shortest Path 3=Rotary Positive 4=Rotary Negative 7=Relative 8=Capture
Set_PositionCommandOverride	BOOL	0=Feature not used 1=Any index executing is terminated and the current index is executed.
Set_PositionCommandOverlap	BOOL	0=Feature not used 1=This index is overlapped (or blended) with an executing index at the end of that executing index (during the deceleration portion)
Set_CapturedPositionSelect	BOOL	This is used with the Capture (CAP) MoveType and specifies which High Speed input is used. 0=DI9 1=DI10

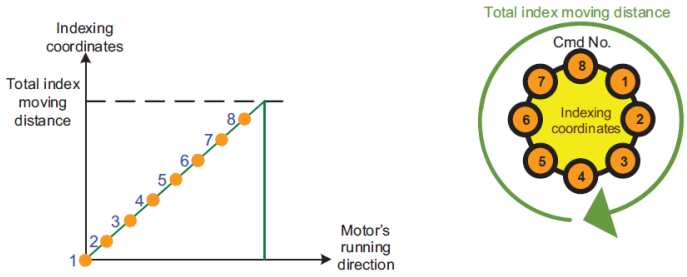
4.5.1 Set_MoveType

Below is an example of the move types. This entry specifies the index type to be executed. In this example, 60,000 is used as *Set_PositionReference*.


Absolute command (ABS) 60000 PUU
Motor's current position (Fb_PUU)
Target position
0 10000 20000 30000 40000 50000 60000 70000 80000 90000 100000
Relative command (REL) 60000 PUU
Motor's current position (Fb_PUU)
60000
Target position
0 10000 20000 30000 40000 50000 60000 70000 80000 90000 100000
Incremental command (INC) 60000 PUU
Motor's current position (Fb_PUU)
Endpoint of previous command (Cmd_E)
60000
Target position
0 10000 20000 30000 40000 50000 60000 70000 80000 90000 100000
High-speed position capturing (CAP) 60000 PUU
Capturing position
60000
Motor's current position (Fb_PUU)
Target position
0 10000 20000 30000 40000 50000 60000 70000 80000 90000 100000

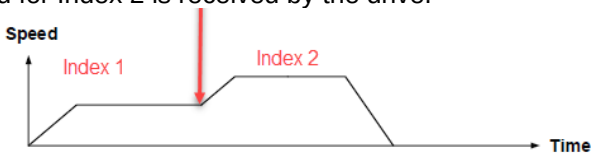
The Rotary move types are used to provide a way to index while observing the natural rollover of the feedback device. For example, if the motor could only index positive, the Rotary Positive is used. When the feedback device transitions through its natural unwind (typically 2.1billion counts), the movements will always index positive.
IMPORTANT: At this time, the Kinetix 5100 does not have an Unwind function. The rotary selections in this AOI do not refer to rotary axis types.





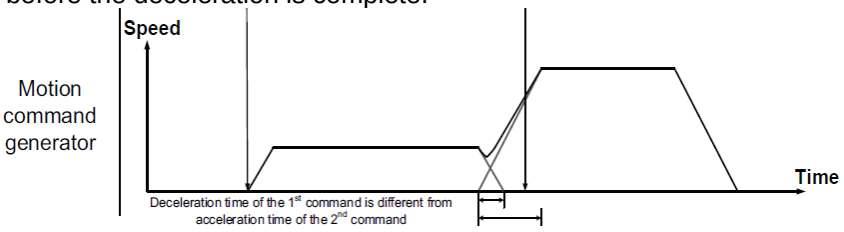
4.5.2 Set_PositionOverride

The executing index (Index 1) is terminated. The new index (Index 2) is executed using its dynamics. This is shown in the graphic below. The red arrow is the point where the command for Index 2 is received by the drive.

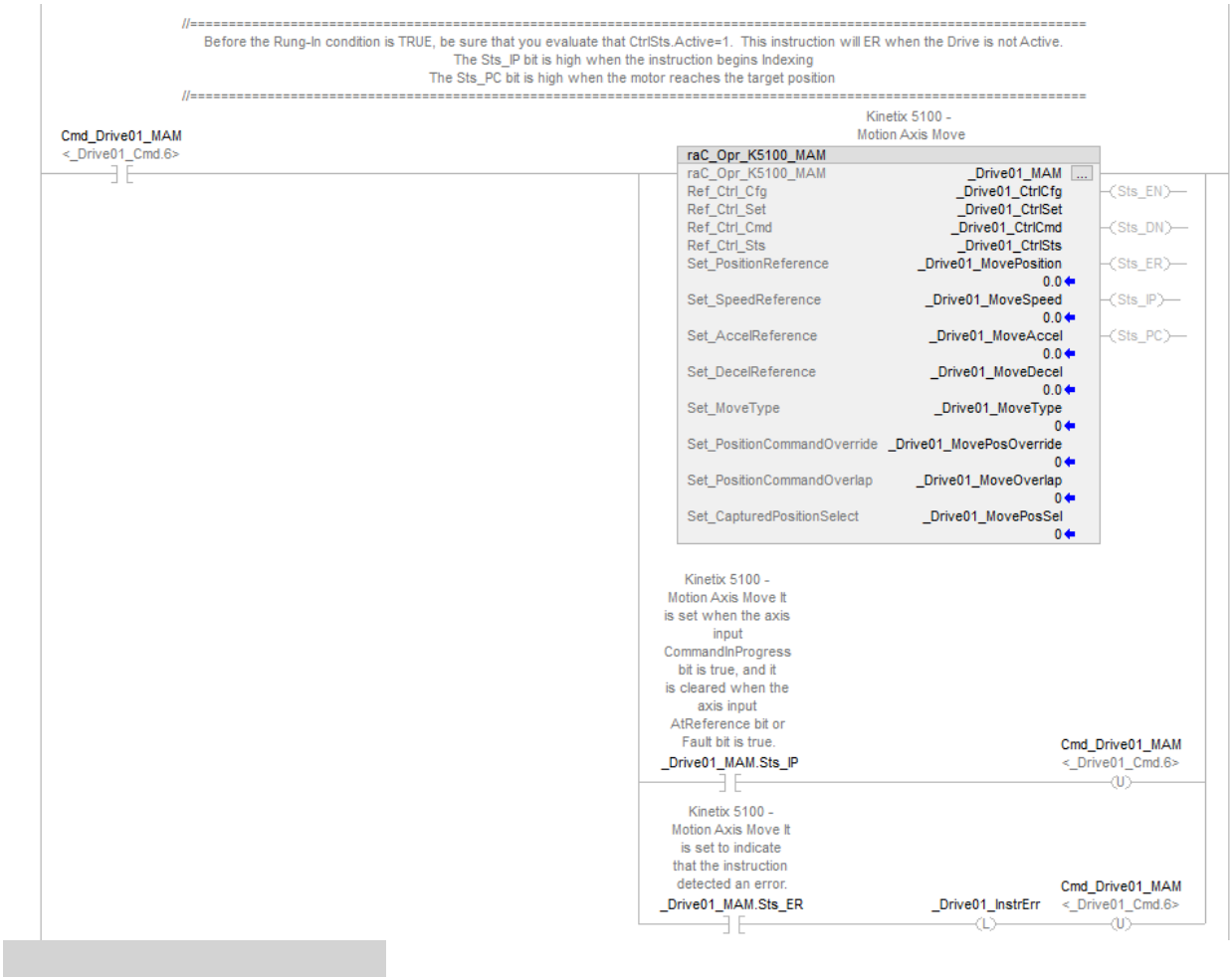


4.5.3 Set_PositionCommandOverlap

The executing index is interrupted during its deceleration. The new index is started before the deceleration is complete.

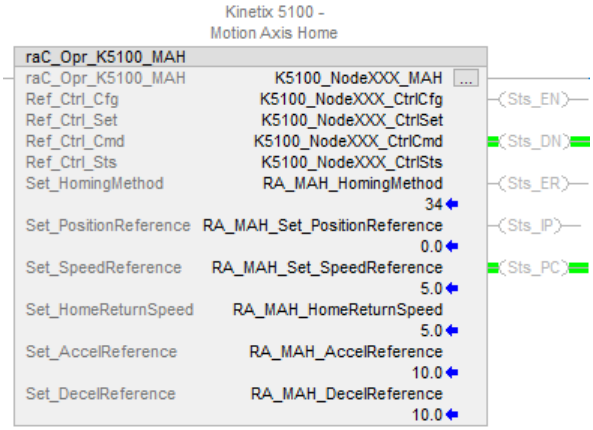

Motion command generator
Speed
Time
Deceleration time of the 1st command is different from acceleration time of the 2nd command

This example shows the Motion Operation AOI (raC_Opr_K5100_MAM). The rung-in condition is reset once the motor is moving to its target position.


//==========
Before the Rung-In condition is TRUE, be sure that you evaluate that CtrlSts.Active=1. This instruction will ER when the Drive is not Active.
The Sts_IP bit is high when the instruction begins Indexing
The Sts_PC bit is high when the motor reaches the target position
//==========
Cmd_Drive01_MAM
<_Drive01_Cmd.6>
Kinetix 5100 -
Motion Axis Move
raC_Opr_K5100_MAM
raC_Opr_K5100_MAM _Drive01_MAM
Ref_Ctrl_Cfg _Drive01_CtrlCfg
Ref_Ctrl_Set _Drive01_CtrlSet
Ref_Ctrl_Cmd _Drive01_CtrlCmd
Ref_Ctrl_Sts _Drive01_CtrlSts
Set_PositionReference _Drive01_MovePosition 0.0
Set_SpeedReference _Drive01_MoveSpeed 0.0
Set_AccelReference _Drive01_MoveAccel 0.0
Set_DecelReference _Drive01_MoveDecel 0.0
Set_MoveType _Drive01_MoveType 0
Set_PositionCommandOverride _Drive01_MovePosOverride 0
Set_PositionCommandOverlap _Drive01_MoveOverlap 0
Set_CapturedPositionSelect _Drive01_MovePosSel 0
Sts_EN
Sts_DN
Sts_ER
Sts_IP
Sts_PC
Kinetix 5100 -
Motion Axis Move It
is set when the axis
input
CommandInProgress
bit is true, and it
is cleared when the
axis input
AtReference bit or
Fault bit is true.
_Drive01_MAM.Sts_IP
Cmd_Drive01_MAM
<_Drive01_Cmd.6>
U
Kinetix 5100 -
Motion Axis Move It
is set to indicate
that the instruction
detected an error.
_Drive01_MAM.Sts_ER
_Drive01_InstrErr
L
Cmd_Drive01_MAM
<_Drive01_Cmd.6>
U

4.6 raC_Opr_K5100_MAH - Motion Axis Home

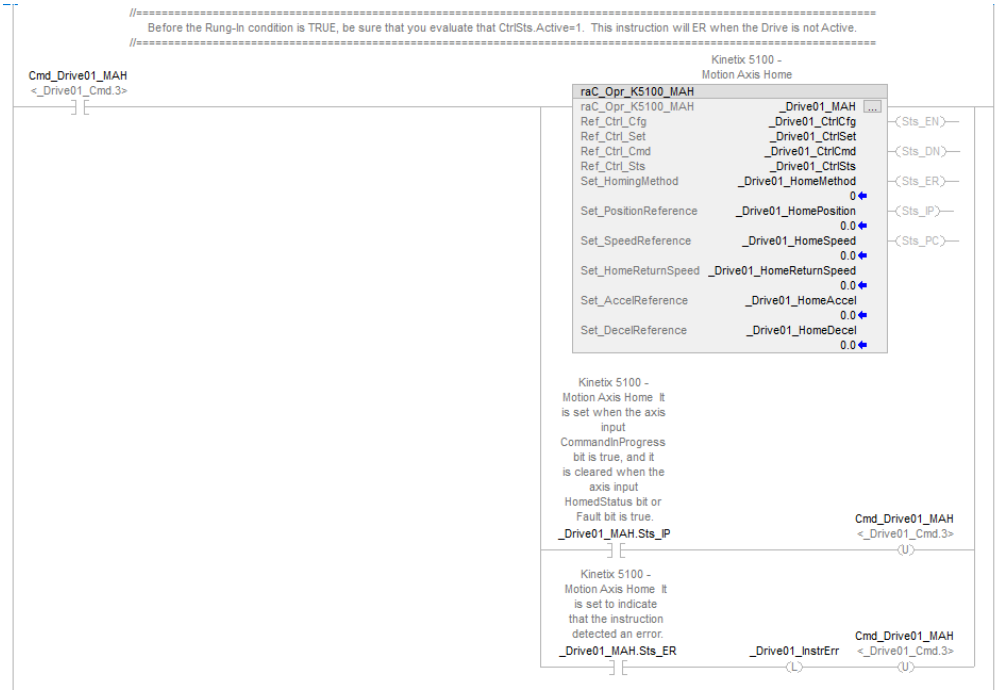
Use the Motion Axis Home AOI to command a Homing Procedure in the drive. Homing is used to define an origin for your motor and to establish an absolute positioning reference for the motor. Once homing is complete, the *Sts.Homed* bit in the CtrlSts UDT is set.



Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to Home is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to Home the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the Home message transaction is successful, and the homing begins. This bit will remain set if the homing is executing.
Sts_PC	BOOL	Process Complete. This bit is set when the rung makes a false-to-true transition and the Homing Sequence is completed.
Ref_Ctrl_Cfg	raC_UDT_Ift_K5100_Cfg	Use the associated Device Object Configuration control interface for this AOI
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI

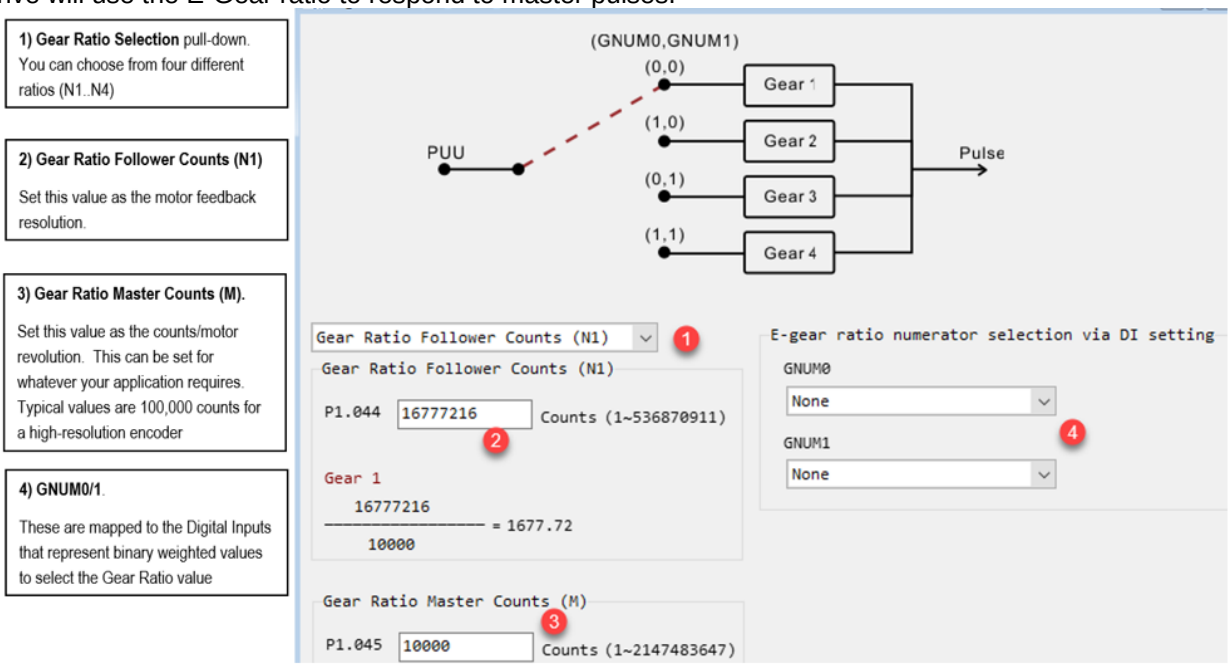
Set_HomingMethod	SINT	Set this value for the Homing Method used with your drive. These entries can be found in the Kinetix 5100 User Manual, Appendix C. They are also listed when you use F1 on the MAH_AOI.
Set_PositionReference	REAL	The home position the drive will use.
Set_SpeedReference	REAL	The Homing Speed used with the MAH (when required by the HomingMethod). Units: 0.1 RPM, Range: 1…20,000; Position Units are converted into RPM using the Device Object AOI
SetHomeReturnSpeed	REAL	The Low-Speed homing setting (when required by HomingMethod) Units: 0.1 RPM; Range: 1…5000; Position Units are converted into RPM using the Device Object AOI
Set_AccelReference	REAL	Accel rate used by HomingMethod. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI
Set_DecelReference	REAL	Decel rate used by HomingMethod. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI

This example shows the Motion Operation AOI (raC_Opr_K5100_MAH). The rung-in condition is reset once the Homing Sequence is in process. Depending on the Homing Method, the homing sequence may complete away from the Home Position specified. This is because of the Homing Deceleration. You may need to execute a raC_Opr_K5100_MAM Absolute type back to the Home Position.



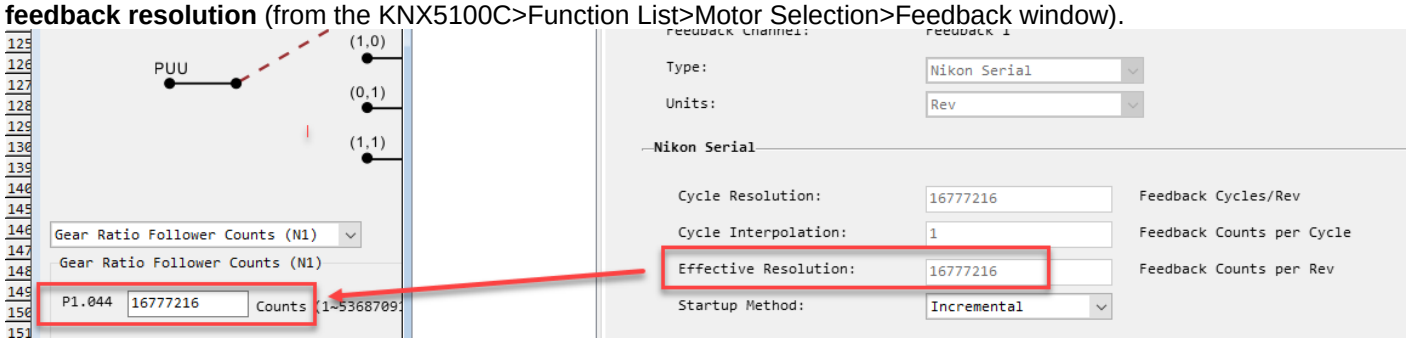
4.7 raC_Opr_K5100_MAG - Motion Axis Gear and E-Gear Ratio

Use the Motion Axis Gear AOI to execute a pulse-pulse relationship with the drive. The MAG AOI uses the E-Gear ratio configured in the KNX5100C software. The E-Gear ratio window is shown below. When the MAG AOI is used, the drive behaves like it is in PT (Position Terminal – or Pulse Train) mode and the drive will use the E-Gear ratio to respond to master pulses.



The PT Mode is a pulse-pulse relationship. When the variable ratio is used and the ratio is changed, there is NO positioning ability. This means when you are finished using the MAG AOI, your position scaling (which also uses the E-Gear ratio) may have changed if you used variable GearingMode and changed the Master ratio because of your application requirements.

The MAG **SetSlaveCounts** is sometimes called the numerator (Shown as 2 above). This is because when you look at how it is used in the drive, it is used to determine the drive's internal 'ratio' (shown as 1677.72 above). For our purposes**, the E-Gear ratio Follower = MAG SetSlaveCounts = motor feedback resolution** (from the KNX5100C>Function List>Motor Selection>Feedback window).



The MAG **Set_MasterCounts** is sometimes called the denominator (shown as 3 above). Any gearing relationship must consider the actual motor mechanics, like a gearbox, actuator pitch, etc. and use this to relate back to a motor rotation. Gear Ratio Master counts is ***desired counts* / motor rotation.** *Desired counts* are not used for positioning; but defines how many counts your motor will move in one rotation based on the number of feedback pulses you expect to receive from the source input; this is used to determine your gearing relationship. So, this Master counts value is used to define the pulse-pulse

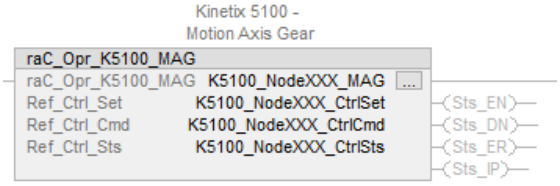
relationship.

IMPORTANT: The MAG AOI can affect your positioning. The issuing Kinetix 5100 (slave) uses the E-Gear ratio to define how it follows pulses from a source (a master). While the result is that the issuing Kinetix 5100 (slave) follows pulses from another source (master), the way the function operates can affect positioning of the drive. Regardless of Operation Mode, the E-Gear ratio is always used to provide a representation of positioning (units or counts) or to define a Pulse-Pulse Following relationship (MAG/PT). When the E-Gear ratio is changed, the positioning of the axis is changed.

Gearing Example:

The master in our system is a 4000 ppr encoder. This means, when the encoder makes one revolution, we expect the Slave1 drive to see: 4000 pulses.
Our application requirement is that we want to follow this encoder at a 1:2 relationship. This means when the master encoder moves 1 encoder revolution, the motor rotates 2 times.
The Master PPR is not entered anywhere, but is required that we know this value. We calculate the MAG SetMasterCounts value knowing the Master PPR counts and the relationship we want in Slave1's motor.
We set the MAG **SetSlaveCounts = Motor Feedback Resolution =16,777,216**
When we set MAG **Set_MasterCounts = 2,000**, this means as the Slave1 drive sees 2000 master pulses, the motor moves 1 rotation, and thus, as the Master encoder moves 4,000 pulses, Slave1 would have moved 2 rotations.

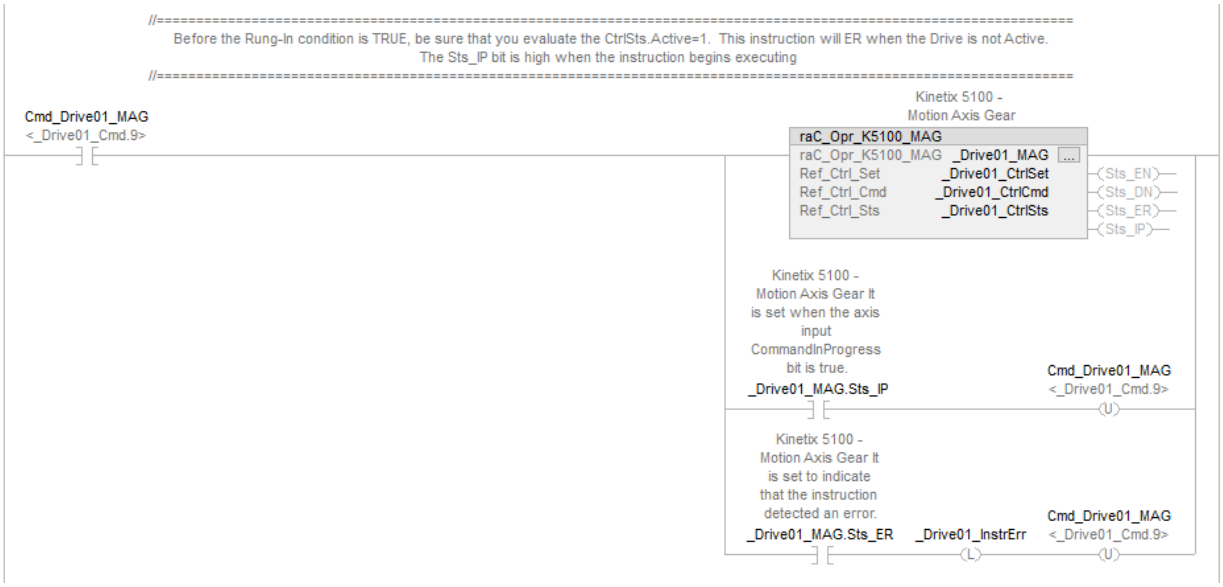
There are two modes of the MAG function that can be used. These are defined by the Cfg_GearingMode entry. This is not visible and is set for Fixed initially. The user must intentionally change this setting. Fixed mode will not impact positioning since it uses the existing E-Gear ratio in the Kinetix 5100. This means that we can follow a master source at this fixed ratio and when gearing is disabled, we can continue positioning without losing the position scaling for the drive.
Variable mode means we can change the E-Gear ratio by manipulating the master/slave counts values. This changes the ratio which the issuing drive (slave) follows the master. However, the positioning is affected since the E-Gear ratio defines the position scaling. This is the same as using the Kinetix 5100 PT sub mode of operation. This mode is useful when your application must follow a Master with different ratios and positioning is not important; or if you can issue a Homing Sequence to re-establish an origin when gearing is completed.



Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to MAG is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to MAG (Sts_EN) is complete.

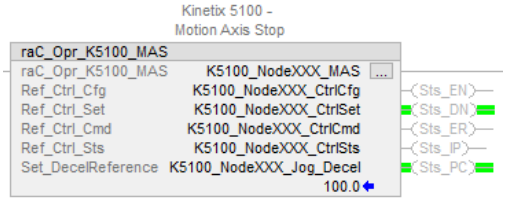
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the MAG message transaction is successful, and the drive begins following. This bit will remain set as the motor is executing the gearing. It remains set while the MAG is active, regardless of the rung-in condition
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI
Cfg_GearingMode	BOOL (Not visible)	0=Fixed 1=Variable
Set_AccelReference	REAL (Not visible)	Accel rate used for the MAG. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI
Set_MasterCounts	DINT (Not visible)	the ***desired counts* / motor rotation.** *Desired counts* are not used for positioning; but defines how many counts your motor will move in one rotation based on the number of feedback pulses you expect to receive from the source input. Range:1-536870911 counts
Set_SlaveCounts	DINT (Not visible)	Set to the motor feedback resolution. Range:1-2147483647

This example shows the Motion Operation AOI (raC_Opr_K5100_MAG). The rung-in condition is reset once the Gearing is in process.



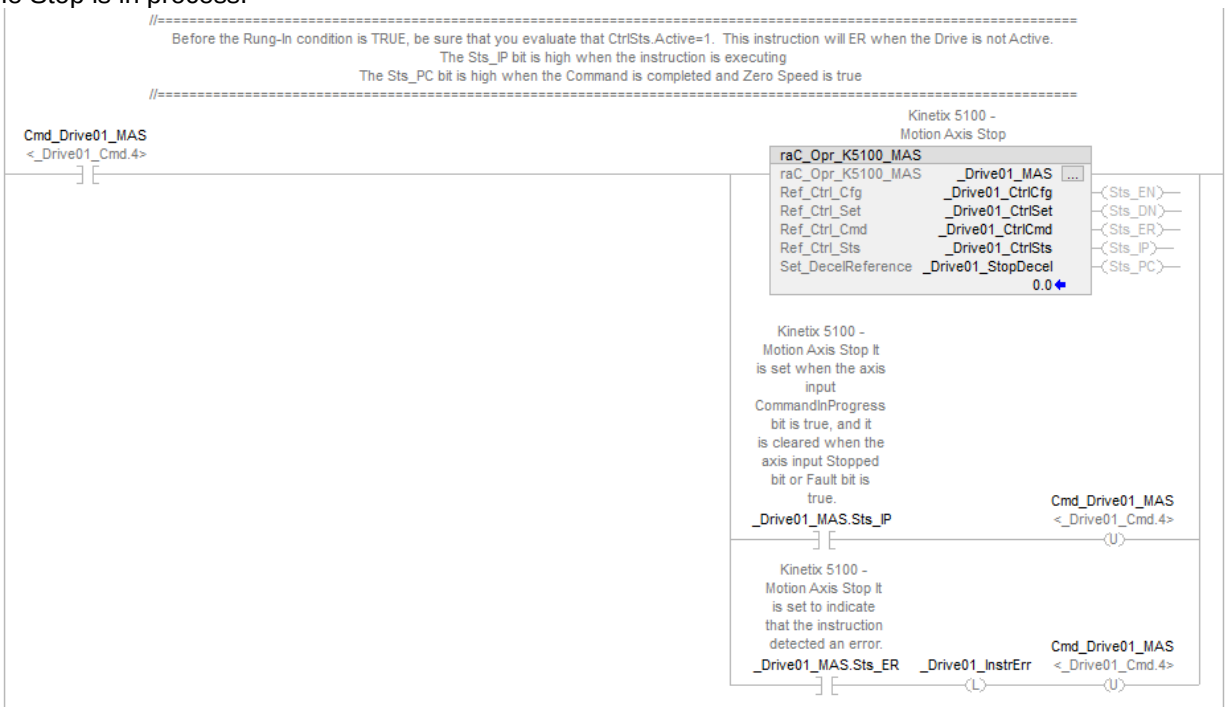
4.1 raC_Opr_K5100_MAS - Motion Axis Stop

Use the Motion Axis Stop to command the drive to stop any motion. The drive remains active when the stop function is complete.



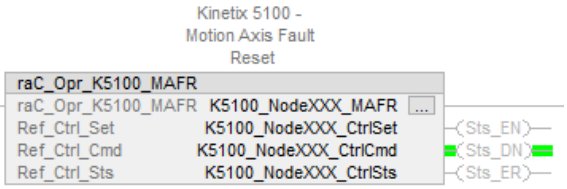
Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to Stop is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to Stop the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the Stop message transaction is successful, and the motor begins to decelerate. This bit will remain set as the motor is executing the stop.
Sts_PC	BOOL	Process Complete. This bit is set when the rung makes a false-to-true transition, the Sts_IP is set, and Zero Speed is reached. Zero Speed is defined using KNX5100C software>General Setting
Ref_Ctrl_Cfg	raC_UDT_Ift_K5100_Cfg	Use the associated Device Object Configuration control interface for this AOI
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI
Set_DecelReference	REAL	Decel rate used for the jog. Units: 0.1 RPM/s, Range:458…30,000,000; Position Units are converted into RPM/s using the Device Object AOI

This example shows the Motion Operation AOI (raC_Opr_K5100_MAS). The rung-in condition is reset once the Stop is in process.



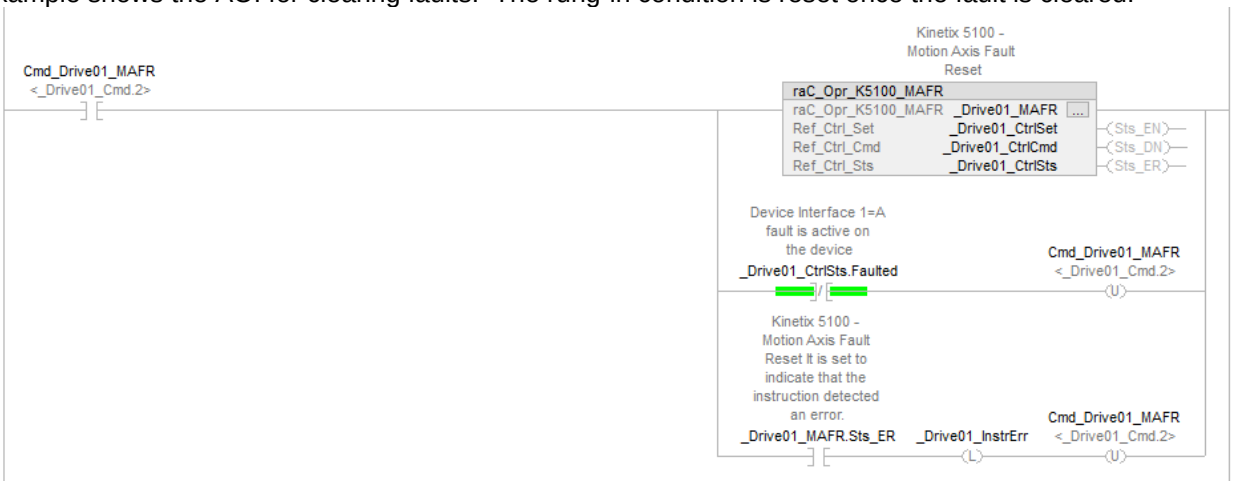
4.2 raC_Opr_K5100_MAFR - Motion Axis Fault Reset

Use the Motion Axis Fault Reset to clear drive faults. When the fault is no longer active in the drive, this instruction will clear the fault. This will not clear any active alarms.



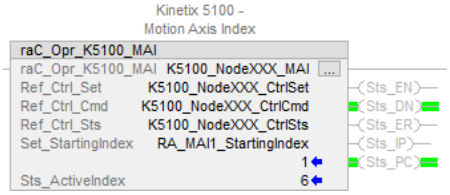
Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to Reset is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to Reset the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI

This example shows the AOI for clearing faults. The rung-in condition is reset once the fault is cleared.



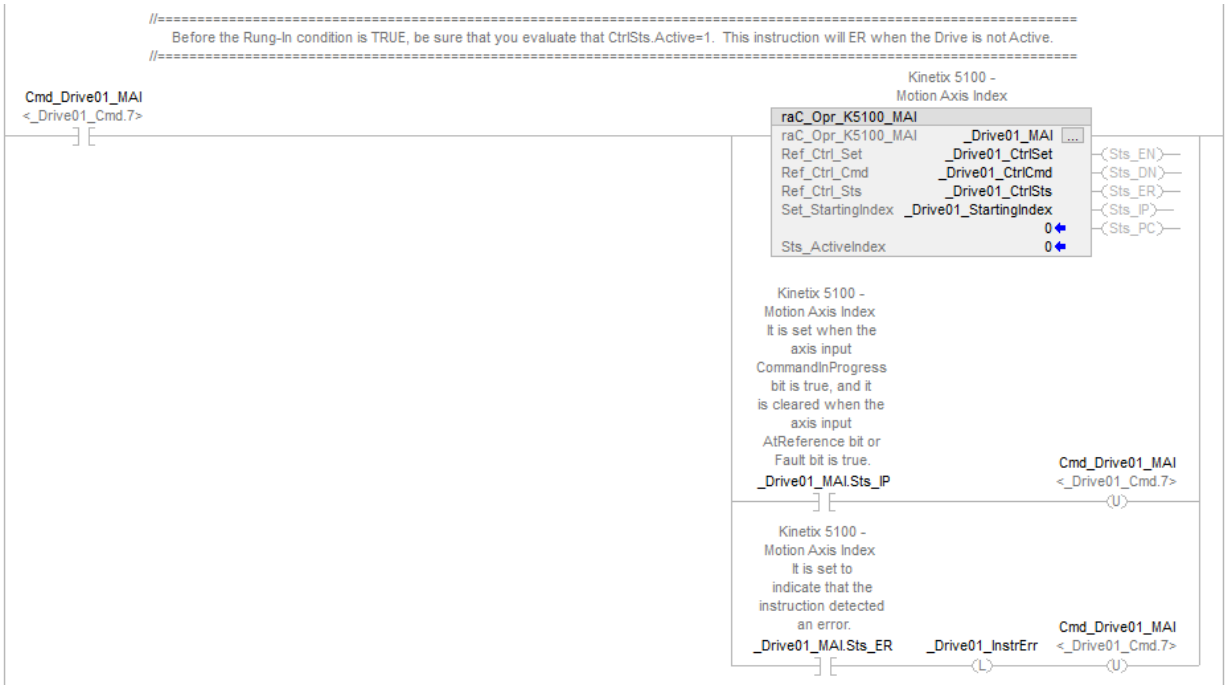
4.3 raC_Opr_K5100_MAI - Motion Axis Index

Use the Motion Axis Index AOI to execute a pre-defined PR (Position Register) in the drive. This Motion Operation AOI will execute any PR type, including indexes, statements, or Parameter read/writes.



Status Bit	Tag Type	Description
Sts_EN	BOOL	Enable. This bit is set when the rung makes a false-to-true transition and the message transaction to MAI is initiated and in process. It remains high until the rung-in condition is false and no faults are active.
Sts_DN	BOOL	Done. This bit is set when the rung makes a false-to-true transition and the message transaction to MAI the drive (Sts_EN) is complete.
Sts_ER	BOOL	Error. This bit is set when the rung makes a false-to-true transition and there is an error that has occurred with the instruction. (This instruction error can be as a result of a fault on the drive itself). See Sts_ERR for details on the cause of the error.
Sts_IP	BOOL	In Process. This bit is set when the rung makes a false-to-true transition, the MAI message transaction is successful, and the PR command has been sent to the drive. This bit will remain set until the AtReference bit is set.
Sts_PC	BOOL	Process Complete. This bit is set when the rung makes a false-to-true transition, the Sts_IP is set, and the MAI has sent the PR execution and the AtReference bit is set.
Ref_Ctrl_Set	raC_UDT_Itf_K5100_Set	Use the associated Device Object Settings control interface for this AOI
Reg_Ctrl_Cmd	raC_UDT_Itf_K5100_Cmd	Use the associated Device Object Command control interface for this AOI
Ref_Ctrl_Sts	raC_UDT_Itf_K5100_Sts	Use the associated Device Object Status control interface for this AOI
Set_Starting_Index	INT	The PR selection to execute in the drive
Sts_Active_Index	INT	Shows the recently executed PR in the drive

This example shows the Motion Operation AOI (raC_Opr_K5100_MAI). The rung-in condition is reset once the MAI is in process.



4.4 Common Error Codes

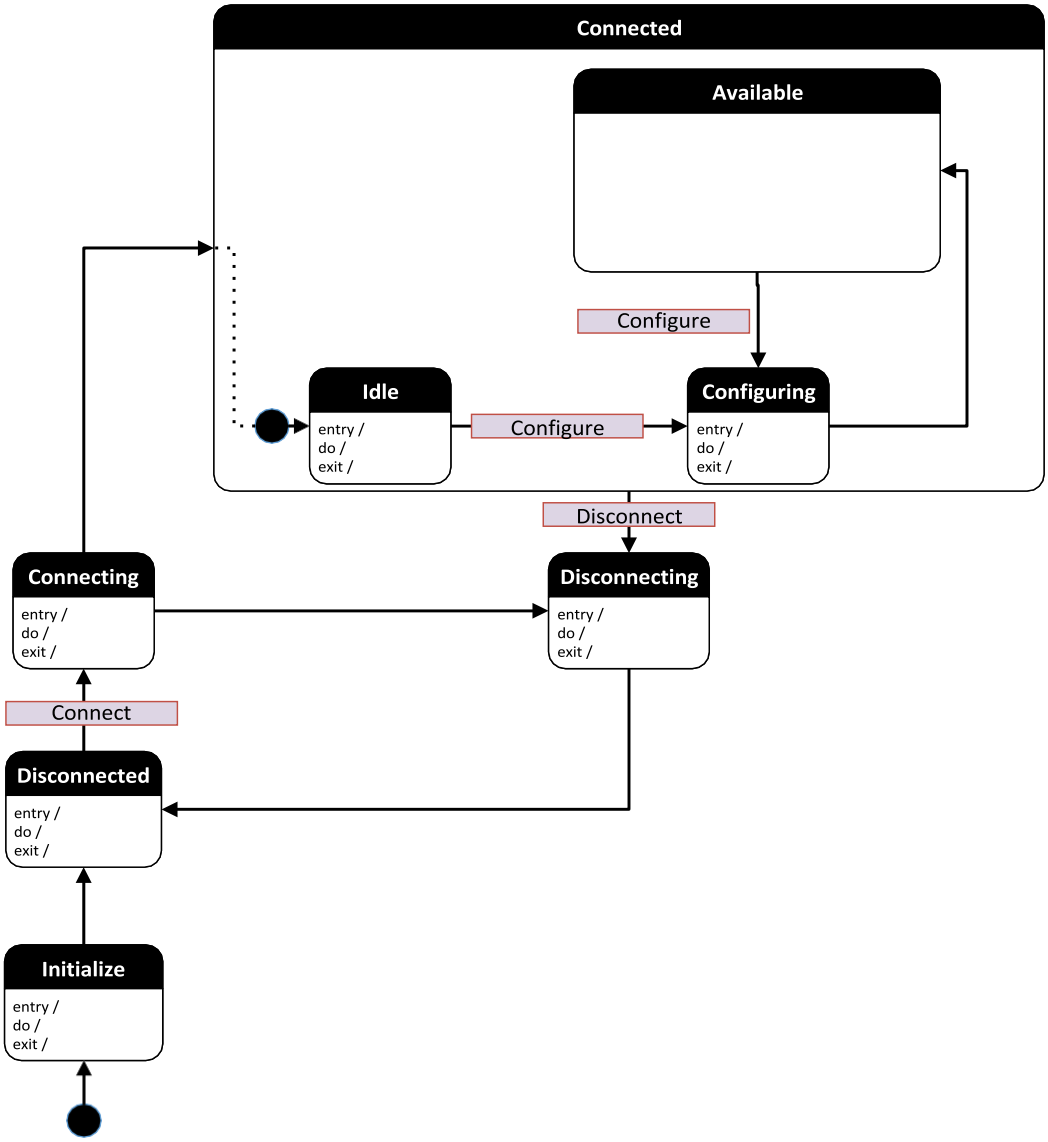
The AOI's use common error codes as much as possible.

Error	Description	Instruction Name									
		MSO	MSF	MAFR	MAS	MAJ	MAT	MAI	MAM	MAH	MAG
100	Drive is not ready	X	X	X	X	X	X	X	X	X	X
101	Drive is faulted	X	X	X	X	X	X	X	X	X	X
102	raC_Opr_K5100_MSO is executing	X									
103	raC_Opr_K5100_MSF is executing	X			X	X	X	X	X	X	X
104	Another raC_Opr_K5100_MSF is executing		X								
105	Drive is disabled				X	X	X	X	X	X	X
106	Another raC_Opr_K5100_MAFR message is executing			X							
107	raC_Opr_K5100_MAS is executing				X	X	X	X	X	X	X
108	Another RA motion Add-On Instructions is sending the command.					X	X	X	X	X	X
111	SpeedReference is out of range					X			X	X	
112	AccelReference is out of range					X			X	X	X
113	DecelReference is out of range				X	X			X	X	
115	StartingIndex is higher than 99							X			
116	Torque is out of range						X				
117	NonCyclicMoveType is higher than 3								X		
118	CyclicMoveType is higher than 2								X		
119	TravelMode is not either 2 or 10								X		
122	HomingMethod is out of range									X	
125	TorqueRampTime is out of range						X				
126	Homing is not completed								X		
127	Previous command has not completed				X	X	X	X	X	X	X
129	Motor is not connected	X	X	X	X	X	X	X	X	X	X
131	Gear slave counts is out of range										X
132	Gear master count is out of range										X
133	Gear ratio is out of range										X
140	Operation is not supported when device is virtual				X	X	X	X	X	X	X
141	Motor type not supported (Linear)				X	X	X	X	X	X	X

5. State Model

The following section will discuss the state model for Device Object. The figure below shows the core logic states.

By default, each state is active for a minimum of 256us to allow for evaluation of state outside of the ADO instance in the user program.



5.1 State Definitions

State	Enumeration	Type	State Command
Initializing	1	Acting	(Initial)
Disconnected	2	Wait	
Disconnecting	3	Acting	Disconnect
Connecting	4	Acting	Connect
Idle	5	Wait	
Configuring	6	Acting	Configure
Available	7	Dual	

5.2 State Actions

State	Action / Function
Initializing	Initialize object and interface (status, command, etc) data structures Set Device Object to operate as Physical Device specific initialization if required
Disconnected	➔ Connect
Disconnecting	
Connecting	Physical - Verify module connection to PAC Virtual – N/A
Idle	➔ Configure
Configuring	Verify all module configuration has been complete.
Available	Device specific

6. Application Code Manager

6.1 Control Asset Objects

These objects contain the AOI definition and any associated UDTs used as linked library to implement object. This gives flexibility to choose to instantiate only definition and create custom implement code. User may also create their own implement library and link with this definition library object.

Member	Description
raC_Dvc_K5100	Kinetix 5100 Device Object
raC_Opr_K5100_MSO	Motion Axis Servo On Operation Object
raC_Opr_K5100_MSF	Motion Axis Servo Off Operation Object
raC_Opr_K5100_MAJ	Motion Axis Jog Operation Object
raC_Opr_K5100_MAT	Motion Axis Torque Operation Object
raC_Opr_K5100_MAM	Motion Axis Move Operation Object
raC_Opr_K5100_MAH	Motion Axis Home Operation Object
raC_Opr_K5100_MAG	Motion Axis Gear Operation Object
raC_Opr_K5100_MAS	Motion Axis Stop Operation Object
raC_Opr_K5100_MAFR	Motion Axis Fault Reset Operation Object
raC_Opr_K5100_MAI	Motion Axis Index Operation Object

6.2 Implement Objects

Implementation Objects contain a configured instance of one or more Control Asset objects and any supplementary data or conditions. That is, the Implementation Object places code on a rung in a specified Program and Routine and creates the backing tags to operate that rung. Additionally, the Implementation Object may include optional components that appear based on user input during configuration. This can include additional rungs of code or conditional statements on those rungs. It may also include things like hardware modules or trend configurations.

6.2.1 raC_LD_Dvc_K5100

Implementation Language: Ladder Diagram

Parameter Name	Default Value	Instance Name	Definition	Description
DeviceRoutine	{ObjectName}	{RoutineName}	Routine	Enter the Routine name where Device object will be inserted
OperationRoutine	{ObjectName}			Enter the Routine name where the Operation instructions will be inserted
TagName	{ObjectName}	{TagName}		Enter the name of the Device Object backing tag
TagDescription	{ObjectDescription}		Local Tag	Enter description of backing tag. This will also appear as faceplate title
IncludeMAM	1			Select this parameter to include Move (MAM) operation
IncludeMAI	1			Select this parameter to include Index (MAI) operation
IncludeMAT	1			Select this parameter to include Torque (MAT) operation
IncludeMAG	1			Select this parameter to include Gear (MAG) operation
ModuleName	Module_{ObjectName}			Enter the existing or new Kinetix 5100 Module Name as it appears in the Controller Organizer
IncludeHardware	1			Select Inclusion of Hardware Module definition. True to include, False to exclude

ModuleType	2198-E1004-ERS/A			Select Module type
Address	192.168.1.0			Enter IP Address of the device
ChassisName	Local			
ParentModule	{ChassisName}			Enter Communication adapter name from Controller Organizer that this module resides under. If it resides directly under controller, enter "Local".
SEAssocDisplay				Associated HMI SE Display
MEAssocDisplay				Associated HMI ME Display

Linked Library

Link Name	Catalog Number	Revision	Solution	Category
raC_Dvc_K5100	raC_Dvc_K5100	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MSO	raC_Opr_K5100_MSO	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MSF	raC_Opr_K5100_MSF	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAFR	raC_Opr_K5100_MAFR	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAH	raC_Opr_K5100_MAH	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAS	raC_Opr_K5100_MAS	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAJ	raC_Opr_K5100_MAJ	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAM	raC_Opr_K5100_MAM	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAI	raC_Opr_K5100_MAI	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAT	raC_Opr_K5100_MAT	2	(RA-LIB) Device	Kinetix5100
raC_Opr_K5100_MAG	raC_Opr_K5100_MAG	2	(RA-LIB) Device	Kinetix5100

Interface

Interface Name	Linked Library	Revision
raC_Itf_K5100_SA	raC_Dvc_K5100	1

raC_Itf_K5100_SA Interface Members

Member Name	Description
PrgName	Program Name
TagName	Tag Name
ModuleName	Module Name

Attachments

Name	Description	File Name	Extraction Path
V1_raC_Dvc_K5100	View Designer	raC_Dvc_K5100.vpd	{ProjectName}\Visualization\ViewDesigner
V1_raC_Dvc_K5100	Global Object ME	(raC-ME) raC_Dvc-Global.ggfx	{ProjectName}\Visualization\FTViewME\GlobalObjects
V1_raC_Dvc_K5100	Faceplate ME	(raC-ME) raC_Dvc_K5100-Faceplate.gfx	{ProjectName}\Visualization\FTViewME\Displays
V1_raC_Dvc_K5100	Global Object SE	(raC-SE) raC_Dvc-Global.ggfx	{ProjectName}\Visualization\FTViewSE\GlobalObjects
V1_raC_Dvc_K5100	Faceplate SE	(raC-SE) raC_Dvc_K5100-Faceplate.gfx	{ProjectName}\Visualization\FTViewSE\Displays
V1_raC_Dvc_K5100	Images	raC_Dvc_K5100_Images.zip	{ProjectName}\Visualization\Images
V1_raC_Dvc_K5100	Reference Manual	RM-raC_Dvc_5100	{ProjectName}\Documentation

7. User Interface

The faceplate files allow you to quickly load, configure and use preconfigured status and diagnostic displays available for View Designer, Factory Talk View ME Edition and Factory Talk View SE Edition.

In this document, the faceplate pictures (or snapshots) are of View Designer and are similar to Factory Talk View ME and SE versions.

Note:

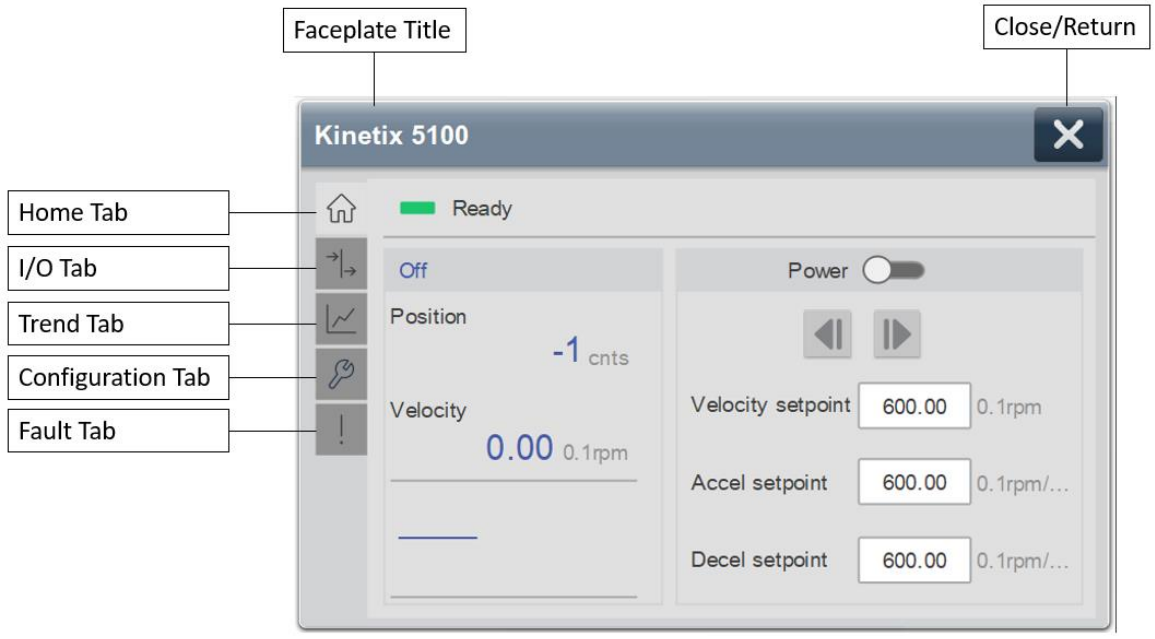
For further information on faceplate configuration and operation refer to the "How_To_Import_and_Configure_Kinetix5100_Device_and Operation_Objects_in_LogixDesigner" and "Operational_Overview_of_CIP_Motion_Faceplates" videos within the Motion folder of the Power Device Library Download files.

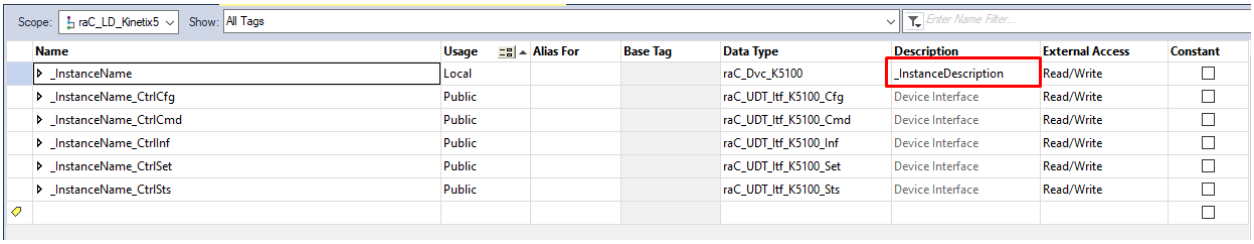
The Power Device Library may be downloaded from the Product Compatibility and Download Center. Search for Power Device Library.
https://compatibility.rockwellautomation.com/Pages/home.aspx

7.1 Faceplate Navigation:

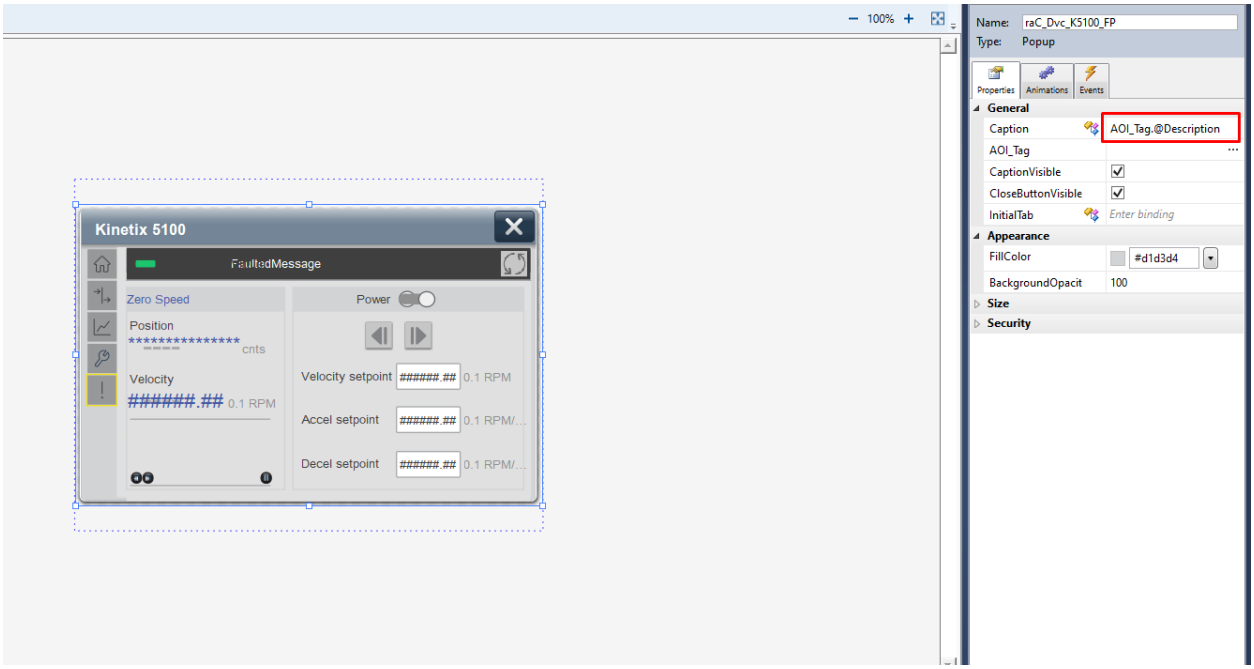
Tab descriptions:

- The **Home** tab provides a drive status summary and basic control functions
- The **I/O** or Input/output tab displays the on/off status of the drive's I/O
- The **Trend** tab displays real-time values of position and velocity for the drive in user units
- The **Config** tab allows the user to configure the trend tab to provide meaningful data
- The **Fault** tab displays a running history of the last (4) faults and alarms with descriptions





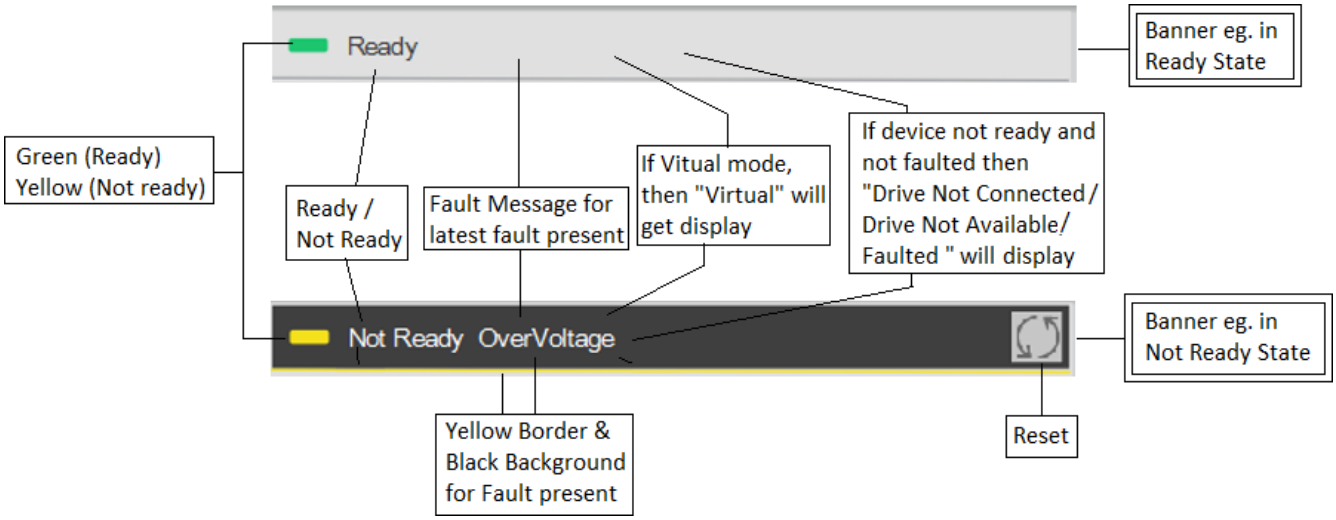
- User can Configure Faceplate Title and Launch button caption from the Description field of the Device Object backing tag in Logix Designer. This tagname will be the name chosen during instantiation or **'InstanceName'**, if defaults were retained.



7.2 Common Status

A banner will be displayed at the top of each screen to provide the following information of the device:

1. Ready or Not Ready
2. Faulted (device will show Not Ready and fault message)
3. Virtualized
4. Communications loss

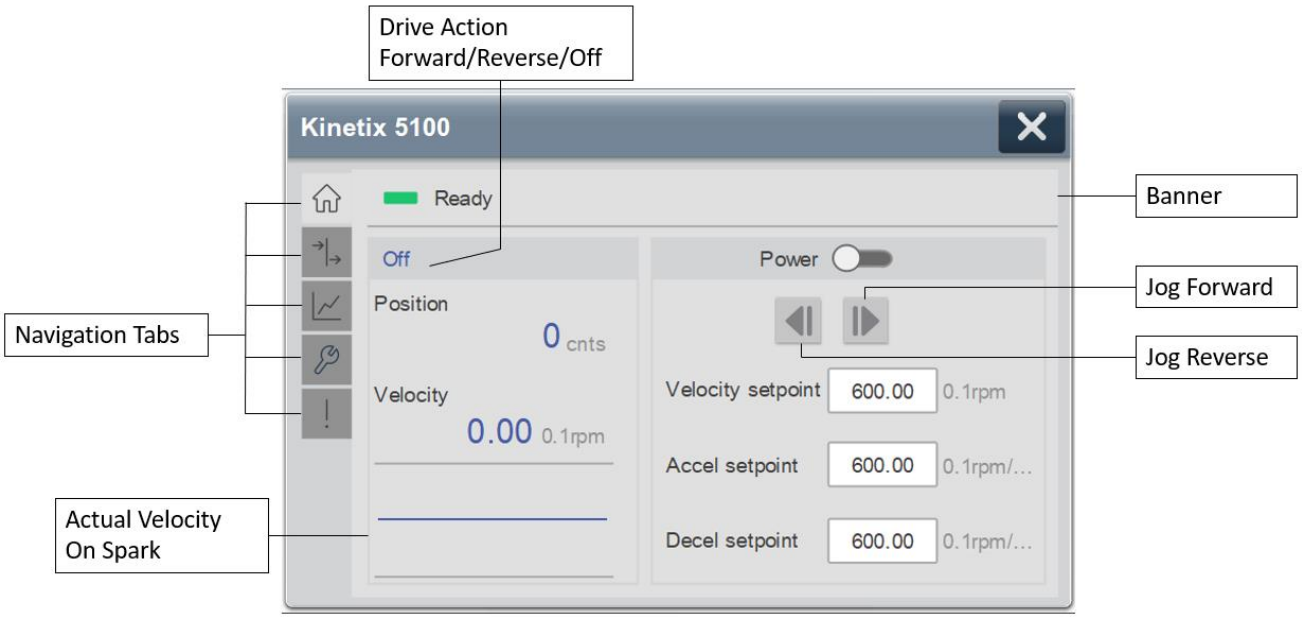

Ready
Banner eg. in Ready State
Green (Ready)
Yellow (Not ready)
Ready / Not Ready
Fault Message for latest fault present
If Vitual mode, then "Virtual" will get display
If device not ready and not faulted then "Drive Not Connected / Drive Not Available/ Faulted " will display
Not Ready OverVoltage
Banner eg. in Not Ready State
Yellow Border & Black Background for Fault present
Reset

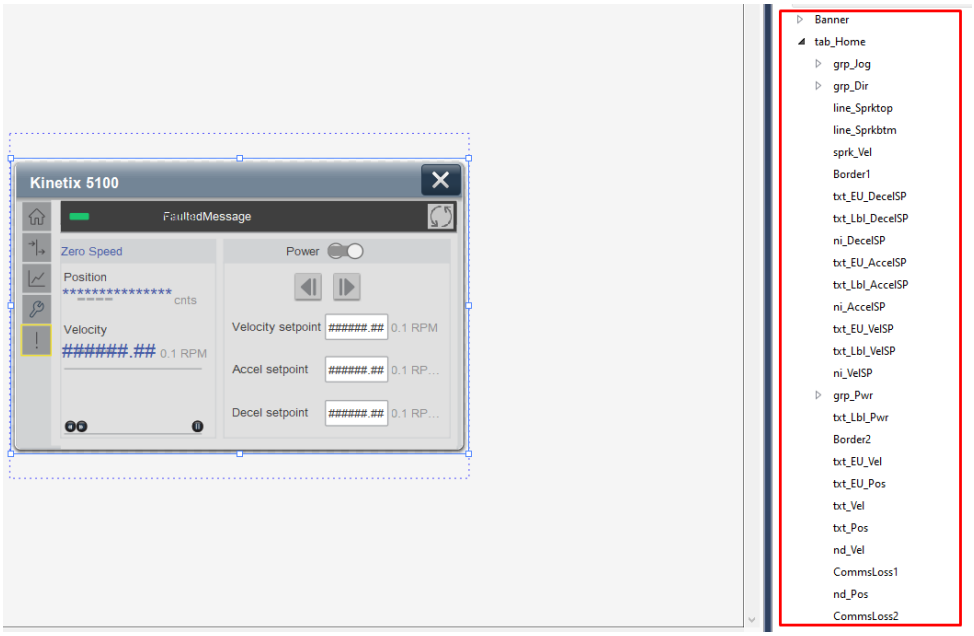
7.3 Home Screen

The Home screen is visible when the Home tab button is selected in the tab control object on the left side of the screen.

The Home screen is the main screen for the faceplate. It contains a Power Group switch object, Group direction indication object, small trends called spark, Position and velocity indication, Jog control object and Velocity, Accel, Decel Set point

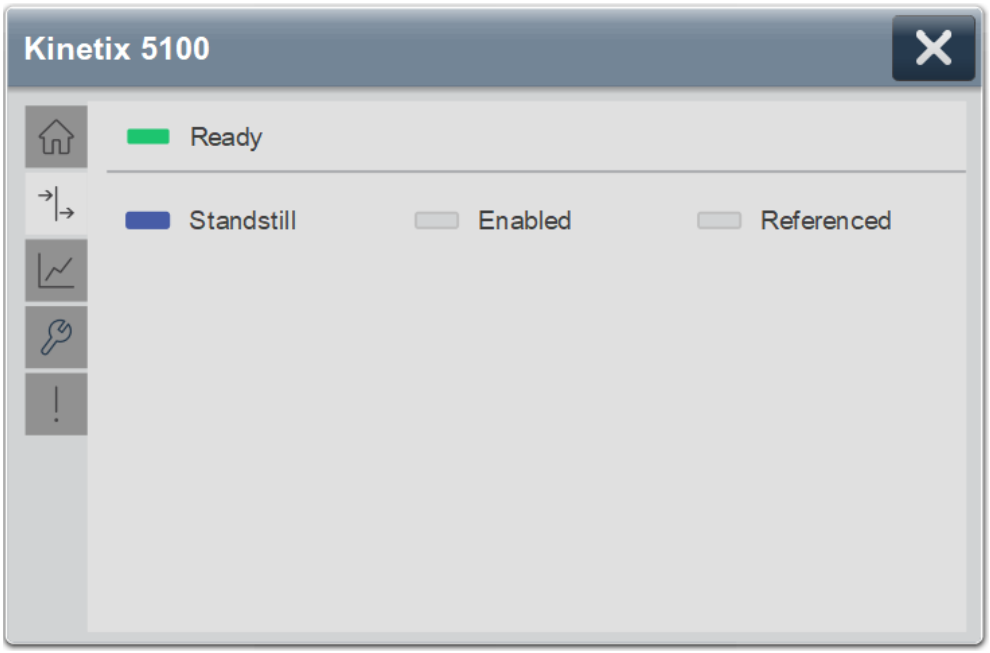
On the top left is a speed multi-state indicator that changes text based on the device state. Other text for state changes include, “Forward”, “Reverse”, “Zero Speed”, and “Off”. The Position and Velocity feedback is a large blue numeric display that is read-only. Below the numeric display is a Spark line to display velocity. There are two push buttons and one toggle button to the right of the display for Power on/off and Jog forward and reverse. Below that, three numeric inputs for the user to set the Jog setpoint, Acceleration setpoint and Deceleration setpoint.





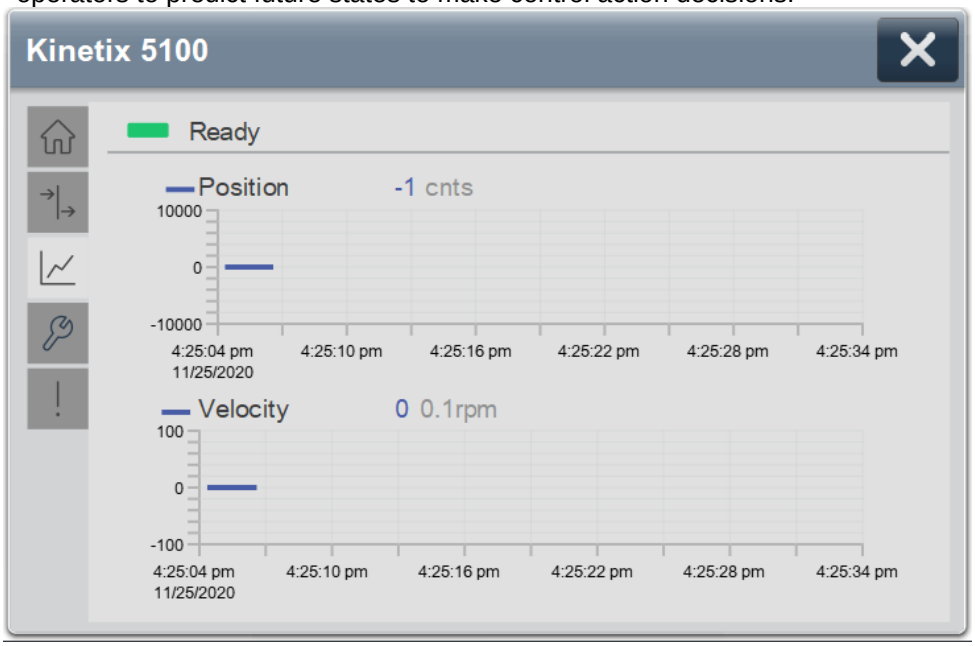
7.4 I/O Screen

The I/O tab displays the on/off status of the drive's I/O. LED shows no color when they are in the OFF position and shows blue when they are in the ON position.



7.5 Trend Screen

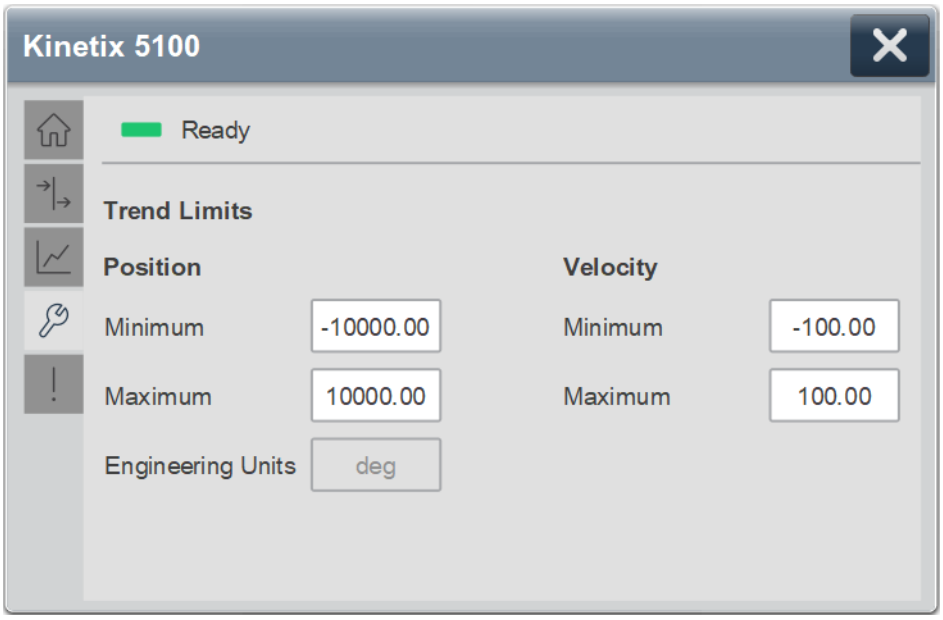
Trends display values over time, often used to compare similar or related values and to allow operators to predict future states to make control action decisions.



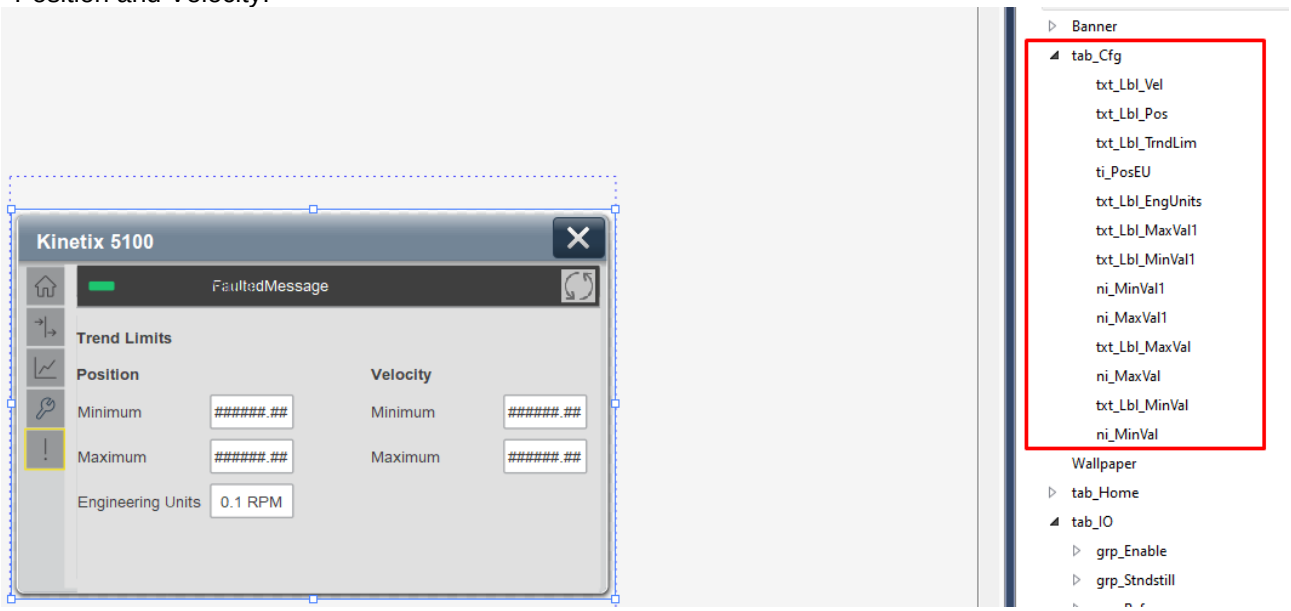
The trend screen displays two trends, Position and Velocity. It contains numeric displays for all three values, labels, and engineering units. The trend also uses its own ready banner that is exactly the same as the “main” ready banner but is shortened in length to accommodate multiple trend charts.

7.6 Configure Screen

The Configure tab contains configuration elements that a maintenance technician would need to troubleshoot and adjust for an object on another tab, i.e., numeric inputs to adjust trend min and max values.

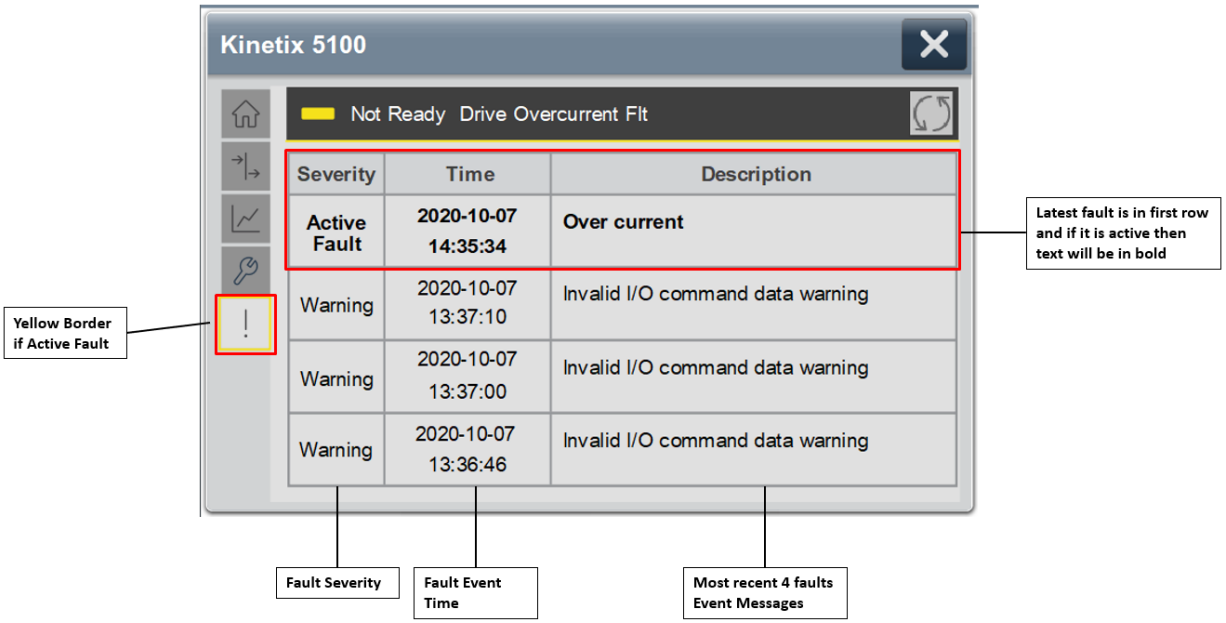


The Configure screen displays contains multiple text labels and four numeric input elements that allows the user to enter the minimum and maximum values to be used on the Trend screen for both Position and Velocity.



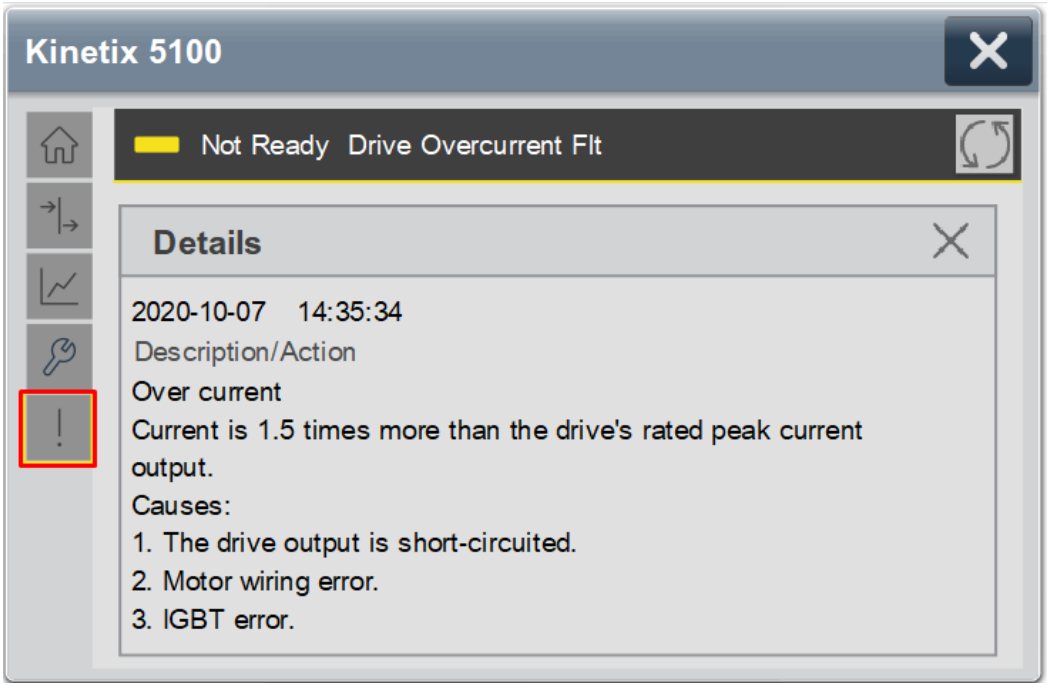
7.7 Fault Warning Screen

The Fault Warning screen displays information for up to four faults for the device. The fault table displays the Severity level (Fault, Warning or Active Fault), time (and date) and a description of the fault.
Note, only row 1 will display the “Active Fault” in the severity column if there is a current active fault, otherwise it will display the last fault. Rows 2-4 only display past faults and warnings, not an active fault.



7.7.1 Details Screen

The Details Screen provides more detailed description and possible action steps to remedy condition.



8. Appendix

For More Configuration and Operational Information

Refer to How to & Operational Videos within the PowerDeviceObjects folder of the Power Device Library download files.

The Power Device Library may be downloaded from the Product Compatibility and Download Center. Search for Power Device Library.
https://compatibility.rockwellautomation.com/Pages/home.aspx

To Provide Feedback or Obtain Additional Information

Contact RASimplification Team at rasimplification@ra.rockwell.com

Disclaimer of Warranty

THE MATERIALS PROVIDED OR REFERENCED BY WAY OF THIS DOCUMENT ARE PROVIDED "AS IS" WITHOUT WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON- INFRINGEMENT OR OTHER VIOLATION OF RIGHTS. ROCKWELL AUTOMATION DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE, VALIDITY, ACCURACY, OR RELIABILITY OF, OR THE RESULTS OF ANY USE OF, OR OTHERWISE RESPECTING, THE MATERIALS PROVIDED OR REFERENCED BY WAY OF THIS DOCUMENT OR ANY WEB SITE LINKED TO THIS DOCUMENT

Limitation of Liability

UNDER NO CIRCUMSTANCE (INCLUDING NEGLIGENCE AND TO THE FULLEST EXTEND PERMITTED BY APPLICABLE LAW) WILL ROCKWELL AUTOMATION BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION, BUSINESS INTERRUPTION, DELAYS, LOSS OF DATA OR PROFIT) ARISING OUT OF THE USE OR THE INABILITY TO USE THE MATERIALS PROVIDED OR REFERENCED BY WAY OF THIS DOCUMENT EVEN IF ROCKWELL AUTOMATION HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IF USE OF SUCH MATERIALS RESULTS IN THE NEED FOR SERVICING, REPAIR OR CORRECTION OF USER EQUIPMENT OR DATA, USER ASSUMES ANY COSTS ASSOCIATED THEREWITH.